WHEN SEMANTIC SEGMENTATION MEETS FREQUENCY ALIASING

Anonymous authors
Paper under double-blind review

ABSTRACT

Despite recent advancements in semantic segmentation, where and what pixels are hard to segment remains largely unexplored. Existing research only separates an image into easy and hard regions and empirically observes the latter are associated with object boundaries. In this paper, we conduct a comprehensive analysis of hard pixel errors, categorizing them into three types: false responses, merging mistakes, and displacements. Our findings reveal a quantitative association between hard pixels and aliasing, which is distortion caused by the overlapping of frequency components in the Fourier domain during downsampling. To identify the frequencies responsible for aliasing, we propose using the equivalent sampling rate to calculate the Nyquist frequency, which marks the threshold for aliasing. Then, we introduce the aliasing score as a metric to quantify the extent of aliasing. While positively correlated with the proposed aliasing score, three types of hard pixels exhibit different patterns. Here, we propose two novel de-aliasing filter (DAF) and frequency mixing (FreqMix) modules to alleviate aliasing degradation by accurately removing or adjusting frequencies higher than the Nyquist frequency. The DAF precisely removes the frequencies responsible for aliasing before downsampling, while the FreqMix dynamically selects high-frequency components within the encoder block. Experimental results demonstrate consistent improvements in semantic segmentation and low-light instance segmentation tasks. The code will be released upon publication.

1 INTRODUCTION

Semantic segmentation is a crucial task in computer vision, with numerous applications such as medical segmentation (Falk et al., 2019), autonomous driving (Hu et al., 2023), robotics (Milioto & Stachniss, 2019), and urban planning (Tong et al., 2020). This dense prediction task heavily relies on high-frequency spatial information that encompasses fine details like textures, patterns, structures, and boundaries (Zhang et al., 2019; Liu et al., 2021a; Zhu et al., 2021; Ding et al., 2019).

While recent models (Cheng et al., 2022; Wang et al., 2023) have achieved much progress, most of these existing methods treat all pixels equally and average each pixel's loss for the image-level one. However, it has long been observed that some pixels are harder to segment than others (Li et al., 2017; Gu et al., 2020). Previous works (Gu et al., 2020; Deng et al., 2022) observe that pixels prone to error are associated with object boundaries, poor lighting conditions, *etc.* Considering this, (Li et al., 2017; Wang et al., 2020) separates an image into two parts: easy and hard regions according to the confidence of predicted probability before applying different segmentation heads. Still, quantitative analysis of 'hard pixels' is an open and challenging problem (Gu et al., 2020).

In this paper, we analyze hard pixels at boundaries and categorize them into three types: 1) false responses, 2) merging mistakes, and 3) displacements. As illustrated in Figure 1, false responses occur when the model predicts boundaries in areas without the object of interest. Merging mistakes refer to the failure to predict boundaries in regions containing the object of interest, resulting in the erroneous merging of two objects into one or missing predictions. Displacements involve predictions that are close to the correct location but deviate from the ground truth object boundary.

We further quantitatively associate three types of errors with a previously unexplored phenomenon: aliasing. Aliasing refers to the overlapping of frequency components in the Fourier domain when a signal is undersampled. According to the Nyquist-Shannon Sampling Theorem (Shannon, 1949;

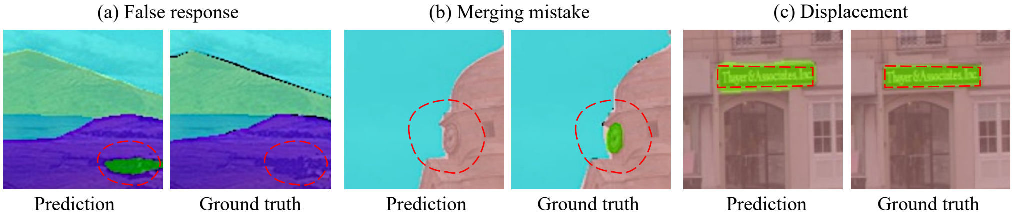

Figure 1: Illustration of three boundary error types.

Nyquist, 1928), signals with frequencies *higher than half of the sampling rate* will be distorted. This is a particularly critical issue in semantic segmentation networks, which typically employ a series of downsampling operations to expand the receptive field and reduce dimensionality (He et al., 2016; Liu et al., 2021b; Wang et al., 2023; Cheng et al., 2022).

Measuring aliasing level needs to calculate the Nyquist frequency. Existing research (Grabinski et al., 2022) calculates Nyquist frequency based on the downsampling stride, *e.g.*, a stride of 2 assumes a sampling rate of $\frac{1}{2}$, causing frequencies above $\frac{1}{4}$ (half of the sampling rate) to become aliased. This assumption holds for point-wise downsampling or downsampling without channel expansion, *e.g.*, a 1×1 convolution, max pooling. However, for larger kernels (*e.g.*, 2×2) and wider output channels (*e.g.*, 2× wider), the actual downsampling rate should be larger, *i.e.*, $\frac{\sqrt{2}}{2}$.

In light of this, we propose to calculate the actual Nyquist frequency based on an equivalent sampling rate, *i.e.*, equivalent Nyquist frequency. The determination of this equivalent Nyquist frequency depends on both the sampling kernel size and the size of the input-output feature maps. Then, we measure the aliasing level with an aliasing score, which is defined as the ratio of high-frequency power above our Nyquist frequency to the total power of the spectrum.

As shown in Figure 2, our analysis reveals a positive correlation between the hard pixels at boundaries and the aliasing score. All three types of errors increase as the aliasing score rises. These errors exhibit distinct characteristics when analyzed from the perspective of aliasing. Additionally, we also note that the importance of these errors varies across different scenarios. For example, displacement errors are primarily concentrated in areas with a high aliasing score, as depicted in Figure 2. In critical tasks such as robotic surgery or radiation therapy, even a displacement of only two or three pixels from vital organs like the brainstem or the main artery can have fatal consequences. In contrast, false responses and merging mistakes, which hold greater significance in autonomous driving scenarios, tend to be more prevalent in areas with relatively low aliasing scores.

The aliasing-induced error can partially explain the success of low-pass filter-based methods (Zhang, 2019; Zou et al., 2020; Grabinski et al., 2022; Chen et al., 2023), as they reduce specific high-frequency components to alleviate aliasing. But they either empirically set the low-pass kernel or underestimate the frequency threshold. Here, we propose two simple yet effective solutions: the de-aliasing filter (DAF) and the frequency mixing module (FreqMix). Based on our calculation of the equivalent Nyquist Frequency, DAF accurately removes the frequencies responsible for aliasing in the Fourier domain during specific downsampling (*e.g.*, 3×3 convolution with stride of two). Given the importance of high-frequency components in representing crucial detailed information (Li et al., 2020a; Bo et al., 2023), FreqMix can dynamically select and balance high-frequency components in each encoder block (*e.g.*, ResNet block). Our main contributions can be summarized as follows:

- Going beyond the binary distinction of easy and hard regions, we categorize pixels challenging for segmentation into three types: false responses, missing edges, and displacements. These three error types hold varying levels of significance in different tasks.
- We introduce the concept of equivalent sampling rate for the Nyquist frequency calculation and propose an aliasing score for quantitative measurement of aliasing levels. Our metric effectively characterizes the three types of errors with distinct patterns.
- We design a simple de-aliasing filter to precisely remove aliasing as measured by our aliasing score. Additionally, we propose a novel frequency-mixing module to dynamically select and utilize both low and high-frequency information. These modules can be easily integrated into off-the-shelf semantic segmentation architectures and effectively reduce three types of errors. Experiments demonstrate consistent improvements over state-of-the-art methods in standard semantic segmentation tasks and low-light instance segmentation.

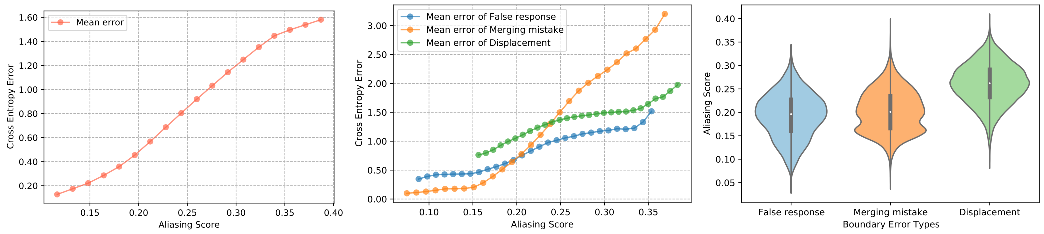

Figure 2: Left: The correlation curve between the cross-entropy error of boundary pixels and the aliasing score. Middle: Cross-entropy error curve for three types of boundary pixels, correlated with aliasing score. Right: Distribution of three types of hard pixels.

2 RELATED WORKS

Hard pixels. Early researches (Shrivastava et al., 2016; Lin et al., 2017b) have found significant variations in classifying samples within images for object detection. Online Hard Example Mining (OHEM) (Shrivastava et al., 2016) selects challenging samples with high losses during training and ignores easily classifiable samples. In contrast, Focal Loss (Lin et al., 2017b) adaptive assigns weights to challenging samples according to their losses. These strategies show effectiveness. Some semantic segmentation approaches (Li et al., 2017) also employ hard sample mining. For instance, Li *et al.* (Li et al., 2017) use shallow or deep networks based on predicted probability scores for easy or hard regions, respectively. Another study (Wang et al., 2020) uses three segmentation heads for coarse segmentation. They process hard and easy regions based on predicted scores and merge the results for the final output. Additionally, methods like Online Hard Region Mining (OHRM)(Yin et al., 2019) and NightLab(Deng et al., 2022) identify challenging regions based on loss values (Yin et al., 2019) or a detection network (Deng et al., 2022). In this work, our finding reveals a positive correlation between the hard pixels and the aliasing score, which is previously unexplored.

Aliasing in Neuron Networks. The aliasing effect in neuron networks is receiving increasing attention. The very first attempt (Zhang, 2019) identified that it is the aliasing makes the neuron network sensitive to the shift. (Zhang, 2019) also proposed the anti-aliasing filters on the convolution layers to enhance shift-invariance. (Zou et al., 2020) introduces learned blurring filters to further improve shift-invariance. Recently, (Hossain et al., 2023) suggests combining a depth adaptive blurring filter with an anti-aliasing activation function. Wavelets transformation (Li et al., 2021) also offers a solution by leveraging low-frequency components to reduce aliasing and enhance robustness against common image corruptions. Beyond image classification, anti-aliasing techniques have gained relevance in the domain of image generation. In (Karras et al., 2021), blurring filters are utilized to remove aliases during image generation in generative adversarial networks (GANs), while (Durall et al., 2020) and (Jung & Keuper, 2021) employ additional loss terms in the frequency space to address aliasing issues. Unlike the empirical design of low-pass filters in existing works, we quantitatively calculate the frequency threshold for aliasing and precisely remove the corresponding high frequencies during downsampling.

Frequency learning Rahaman *et al.* (Rahaman et al., 2019) find the spectral bias that neuron networks prioritize learning the low-frequency modes. Xu *et al.* (Xu & Zhou, 2021) conclude the deep frequency principle, the effective target function for a deeper hidden layer biases towards lower frequency during the training (Xu & Zhou, 2021). Qin *et al.* (Qin et al., 2021) exploring utilizing more frequency for channel attention mechanism. Xu *et al.* (Xu et al., 2020) introduces learning-based frequency selection method into well-known neural networks, taking JPEG encoding coefficients as input. Huang *et al.* (Huang et al., 2023) employs the conventional convolution theorem in DNN, demonstrating that adaptive frequency filters can efficiently serve as global token mixers. Compared to existing works, this work investigates the relationship between high frequency and segmentation error for the first time with introduced aliasing score, demonstrating the possibility of improving segmentation accuracy by optimizing the frequency distribution of features.

3 ALIASING DEGRADATION AND SOLUTION

3.1 ALIASING MEASUREMENT

Irrespective of specific architecture, modern Deep Neural Networks (DNNs) fundamentally consist of a series of stacked convolutional or transformer encoder blocks and downsampling layers (He et al., 2016; Liu et al., 2021b; 2022). From the perspective of signal processing, the downsampling

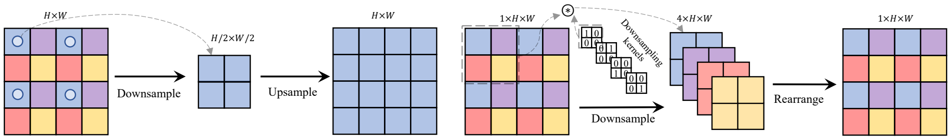

Figure 3: Illustration of how a larger kernel and channel expansion increase the sampling rate. Left: Point-wise downsampling with a stride of 2 and no channel expansion results in a sampling rate of $\frac{1}{2}$, causing aliased high frequencies to be misrepresented as low frequencies. Right: Downsampling using 2×2 identity kernels with 4× channel expansion actually increases the sampling rate to 1, instead of $\frac{1}{2}$, as all pixels are sampled.

operations are prone to violate Nyquist Sampling Theorem (Shannon, 1949; Nyquist, 1928) since the frequency of features, especially those close to the boundary, often exceeds the Nyquist frequency. The resulting frequency overlaps, formally known as aliasing, occur when high frequencies above the Nyquist frequency are undersampled, leading to signal distortion or misinterpretation.

Nyquist frequency calculation. To calculate the Nyquist frequency during downsampling, existing research (Grabinski et al., 2022) considers the stride of the downsampling layer. Specifically, for a downsampling layer with a stride of 2, since it reduces the spatial size by half, the sampling rate is $\frac{1}{2}$. According to the Nyquist Sampling Theorem (Shannon, 1949), the Nyquist frequency is half of the sampling rate, *i.e.*, $\frac{1}{4}$. This estimation is accurate for 1×1 point-wise downsampling layers or depth-wise downsampling layers without channel expansion, such as 1×1 convolutions, max/average pooling. However, it may underestimate the sampling rate when downsampling layers employ larger kernel sizes and output channel expansion, as they increase the actual sampling rate, such 2×2 patch merging and 3×3 convolution in (Liu et al., 2022; 2021b; Wang et al., 2023).

To illustrate this phenomenon, we provide a simple example in Figure 3. By learning four different identity kernels and outputting 4× wider output channels, even though the spatial size is 2× downsampled, the spatial information can be losslessly preserved because all pixels are actually sampled, *i.e.*, the sampling rate is 1 instead of $\frac{1}{2}$.

Based on this observation, we consider the kernel size and feature size (channel, height, and width) instead of just the downsampling stride. We introduce an alternative yet straightforward solution, equivalent sampling rate (ESR), to calculate the actual sampling rate as:

$$\mathrm{ESR} = \min(K^{\mathrm{down}}, \sqrt{\frac{C^{\mathrm{out}}}{C^{\mathrm{in}}}}) \times \sqrt{\frac{H^{\mathrm{out}} \times W^{\mathrm{out}}}{H^{\mathrm{in}} \times W^{\mathrm{in}}}}, \tag{1}$$

where C, H, and W represent channel, height, and width sizes. 'in' and 'out' refer to input and output features. We assume equal sampling rates for height and width, calculated by square root, *i.e.*, $\sqrt{\frac{H^{\mathrm{out}}\times W^{\mathrm{out}}}{H^{\mathrm{in}}\times W^{\mathrm{in}}}} = \frac{1}{\mathrm{Stride}}$, which aligns with (Grabinski et al., 2022). $\min(K^{\mathrm{down}}, \sqrt{\frac{C^{\mathrm{out}}}{C^{\mathrm{in}}}})$ reflects the impact of downsampling kernel size K^{down} and channel expansion. For example, with a 2×2 kernel ($K^{\mathrm{down}} = 2$), it can double the sampling rate if the channel expands by a factor of 4 ($\frac{C^{\mathrm{out}}}{C^{\mathrm{in}}} = 4$), accommodating the sampled information. However, with lower channel expansion (*e.g.*, $\frac{C^{\mathrm{out}}}{C^{\mathrm{in}}} = 2$), it limits the sampling increment to $\min(2, \sqrt{2}) = \sqrt{2}$. Specifically, ResNet (He et al., 2016), Swin-Transformer (Liu et al., 2021b), and ConvNeXt (Liu et al., 2022) use a 3×3 residual block (can be regarded as 3×3 convolution (Ding et al., 2021)), 2×2 patch merging with a 2× channel expansion. Thus, the equivalent sampling rate is $\frac{\sqrt{2}}{2}$, and the corresponding Nyquist frequency is $\frac{\sqrt{2}}{4}$.

Aliasing score. After calculating the Nyquist frequency, we can obtain the set of high frequencies larger than the Nyquist frequency that lead to aliasing: $\mathcal{H} = \{(k,l) \mid |k| > \frac{\mathrm{ESR}}{2} \text{ or } |l| > \frac{\mathrm{ESR}}{2}\}$. Thus, we can quantitatively measure the aliasing level by introducing an aliasing score as a metric, which we define as the ratio of frequency power above the Nyquist frequency to the total power in the frequency spectrum.

Specifically, we first transform the feature map $f \in \mathbb{R}^{C\times H\times W}$ into the frequency domain using the Discrete Fourier Transform (DFT), it can be represented as:

$$F(c,k,l) = \frac{1}{H \times W} \sum_{h=0}^{H-1} \sum_{w=0}^{W-1} f(c,h,w) e^{-2\pi j(kh+lw)}, \tag{2}$$

where $F \in \mathbb{R}^{C\times H\times W}$ represents the output array of complex numbers from the DFT. H and W denote its height and width. And the c, h, w indicates the coordinates of feature map f. The frequencies in the height and width dimensions are given by $|k|$ and $|l|$, with k taking values from the set $\{0, \frac{1}{H}, \ldots, \frac{H-1}{H}\}$ and l from $\{0, \frac{1}{W}, \ldots, \frac{W-1}{W}\}$. Consequently, the aliasing score can be formulated as follows:

$$\text{Aliasing Score} = \frac{\sum_{(k,l)\in\mathcal{H}} |F(c,k,l)|^2}{\sum |F(c,k,l)|^2}. \tag{3}$$

3.2 BOUNDARY ERROR TYPE

Although it is widely known that predicting object boundaries accurately is challenging and often results in high errors, there is no existing work that investigates boundary error types in-depth. Here, we further divide the boundary error into three types, false response, missing edges, and displacement, as shown in Figure 1. Through aliasing analysis, we find they show different characteristics.

False response refers to instances where the predicted edges erroneously appear in non-boundary areas, also known as false positives. These false responses are often associated with texture and noise-related issues as reported in previous studies (Lopez-Molina et al., 2013). Notably, as shown in Figure 2, these errors tend to occur in regions characterized by a relatively low aliasing score.

Merging mistake occurs when the predicted results fail to capture the corresponding edges, resulting in false negatives. They are typically linked to low-contrast regions (Lopez-Molina et al., 2013). As depicted in Figure 2, these errors also manifest in areas with a relatively low aliasing score.

Displacement error arises when the predicted edges deviate from their true positions. Such displacements may occur when the image features are misaligned (Li et al., 2020b; Huang et al., 2021). As depicted in Figure 2, these errors tend to manifest in areas with a relatively high aliasing score.

Metrics for three errors. Additionally, we have designed a series of metrics as tools to assess and analyze the error rate of three distinct types of hard pixel errors at boundaries: false response (FErr), merging mistake (MErr), and displacement (DErr):

$$\text{FErr} = \frac{|P_1 - (G_d \cap P_1)|}{|P_1|}, \text{MErr} = \frac{|G_1 - (P_d \cap G_1)|}{|G_1|}, \text{DErr} = 1 - \frac{|(P_d \cap P) \cap (G_d \cap G)|}{|P_d \cap G_d|} \tag{4}$$

where P and G refer to the binary mask of prediction and ground truth, P_1 and G_1 denote the pixels located on the contour line of the binary mask, P_d and G_d represent the pixels located in the boundary region of the binary mask with a pixel width of d. As per the guidelines presented in (Cheng et al., 2021), we set the pixel width of the boundary region to 15 pixels.

Impact of blur filters. Previous work (Zhang, 2019) has explored the use of a simple Gaussian blur to mitigate aliasing in DNNs. However, due to the absence of a metric for evaluating the level of aliasing and the oversight of three distinct types of hard pixel errors at boundaries, the reason why the counterintuitive phenomenon of a simple blur filter improving performance remains unexplained. Here, armed with the tools of aliasing scores and metrics for three distinct types of hard pixel errors, we take a step further. As shown in Table 1, inserting a 3×3 Gaussian filter before downsampling effectively reduces the average aliasing score from 9.4% to 0.27%, thus improving both the overall segmentation mIoU and boundary segmentation BIoU. We observe that the improvements are primarily due to the reduction in displacement errors, consistent with the findings presented in Figure 1. Displacement errors are predominantly concentrated in regions with high aliasing scores, and the application of a blur filter decreases the aliasing score, subsequently reducing the displacement error.

Table 1: Analysis for the impact of blur filters with UPerNet-R50 on Cityscapes validation set. The ↑/↓ indicates that higher/lower values are better.

Blur Kernel	mIoU↑	Boundary		Three type errors			Aliasing Score
		BIoU↑	BAcc↑	FErr↓	MErr↓	DErr↓	
-	77.8	57.4	73.0	**25.4**	**53.3**	27.6	9.4%
3× 3	**78.8**	**58.0**	**73.6**	26.8	53.4	27.0	0.27%
5× 5	78.2	57.4	73.0	25.5	54.4	**26.4**	0.16%
7× 7	78.0	57.4	73.1	27.9	53.6	27.6	0.14%

Further increasing the blur kernel size to 5×5 and 7×7 does not yield additional improvements; instead, it leads to degradation compared to the 3×3 kernel. This occurs because the larger kernel size results in a higher error rate for false responses (FErr) and merging mistakes (MErr), aligning with the observations in Figure 2 that false responses and merging mistakes predominantly exist in areas with relatively low aliasing scores. In summary, an appropriately sized blur kernel improves the model by reducing hard pixel displacement errors. However, larger blur kernels can lead to a higher error rate in false responses and merging mistakes, potentially counteracting the improvements.

Impact of noise. Previous work (Chen et al., 2023) has revealed the negative impact of noise in low-light scenarios. Here, we further analyze the relationship between aliasing and noise.

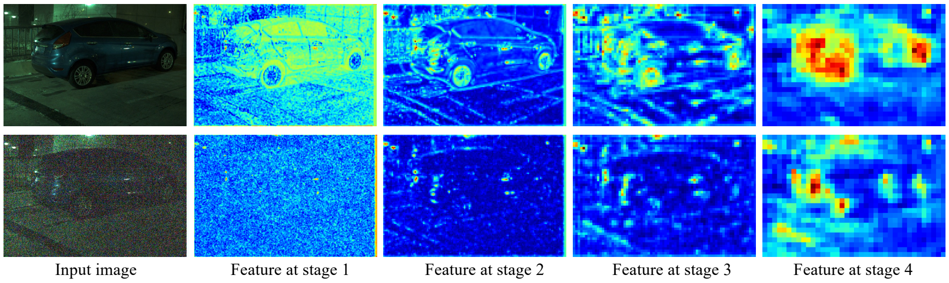

Figure 4: Illustration of the impact of noise on the features at different stages.

Modern backbone models typically organize the convolutional/transformer encoder block into four stages. The first stage is downsampled by a factor of 4 compared to the input images, and there are three 2× downsampling operations from stage-1 to stage-4. Consequently, we present the aliasing score results for the three 2× downsampling operations at stages 1 to 3. As shown in Table 2, we introduce Gaussian noise to images and observe a substantial increase in the aliasing ratio of the feature map at the first stage. This observation indicates that noise amplifies the severity of aliasing effects. However, for deeper features at the second and third stages, the aliasing ratio decreases. This decrease does not signify a reduction in aliasing; rather, it occurs because the earlier high level of aliasing results in severe degradation. Consequently, the features become 'hard pixels,' meaning that the following stage struggles to extract useful information and loses response to objects. Thus feature maps appear plain and lack useful high frequency, as shown in Figure 4. As a result, there is a significant deterioration in all three types of errors, particularly in the case of displacement error.

Table 2: Analysis for the impact of noise with UPerNet-R50 on Cityscapes validation set.

Noise level	mIoU↑	Boundary		Three type errors			Aliasing score		
		BIoU↑	BAcc↑	FErr↓	MErr↓	DErr↓	Stage-1	Stage-2	Stage-3
$\sigma = 0$	**77.8**	**57.4**	**73.0**	**25.4**	**53.3**	**27.6**	5.4%	11.5%	11.3%
$\sigma = 5$	71.2	51.4	64.7	26.2	55.8	35.0	6.1%	11.0%	10.8%
$\sigma = 10$	54.1	37.5	49.0	30.4	62.7	47.0	6.4%	10.3%	9.9%
$\sigma = 20$	20.6	13.4	20.3	55.9	78.7	73.3	9.6%	8.4%	6.4%

3.3 ALIASING-AWARE SOLUTION

In this section, we introduce two solutions: De-Aliasing Filter (DAF) for removing aliasing during downsampling and Frequency Mixing (FreqMix) for adjusting frequency within encoder block.

De-aliasing filter. Several prior techniques, as introduced in (Zhang, 2019; Zou et al., 2020; Hossain et al., 2023), employ methods to attenuate high-frequency components within feature maps prior to downsampling, aiming to prevent aliasing artifacts. These approaches typically employ traditional spatial domain blurring operations for this purpose. Recent work (Grabinski et al., 2022) also explores the removal of high frequencies in the Fourier domain. However, it determines the low-pass cutoff frequency solely based on the downsampling stride, leading to an underestimation of the sampling rate. This results in an excessive removal of high frequencies that are crucial for predicting segmentation details.

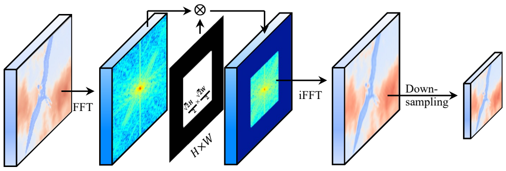

Figure 5: Illustration of de-aliasing filter (DAF). FFT: Fast Fourier Transform, iFFT: Inverse FFT.

On the basis of our Nyquist frequency calculation in Section 3.1, we propose the De-Aliasing Filter (DAF). It first transforms the feature map to be downsampled into the Fourier domain and then sets the power of high frequencies that lead to aliasing to zero. Leveraging the equivalent sampling rate equation 1, we can calculate the actual Nyquist frequency, which serves as the low-pass cut-off frequency for the DAF. It accurately removes the frequency power causing aliasing. This precise removal of frequency power effectively eliminates aliasing. Figure 5 illustrates its procedure in detail. In the spatial domain, this operation would involve convolving the feature map with an infinitely large non-bandlimited filter, which cannot be implemented in practice (Grabinski et al., 2022). Note that DAF is a parameter-free and easily integrable module.

Frequency Mixing Module. The degradation introduced by aliasing underscores the importance of frequency balancing in segmentation. On one hand, high frequencies are crucial for representing important detailed information, such as object boundaries (Li et al., 2020a; Bo et al., 2023). On the other hand, high frequencies above the Nyquist frequency can lead to harmful aliasing. Motivated by this observation, we shift our focus to the feature extraction blocks and introduce a frequency-mixing module. We insert this module after the convolutional layer, where it adaptively weights the frequency power both below and above the Nyquist frequency. Its formal representation is:

$$f'(c,h,w) = A^{\downarrow}(c,h,w) * f^{\downarrow}(c,h,w) + A^{\uparrow}(c,h,w) * f^{\uparrow}(c,h,w), \tag{5}$$

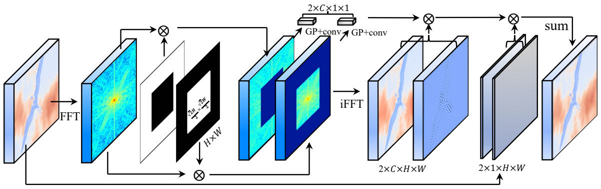

Figure 6: Illustration of Frequency Mixing (FreqMix) module.

where c, h, and w are indices in the channel, height, and width dimensions, and f and f' are input and output feature maps. $f^{\downarrow}$ and $f^{\uparrow}$ are frequency components below and above the Nyquist frequency, which can be simply obtained by low-pass/high-pass filtering in the Fourier domain, as shown in Figure 6. $A^{\downarrow}$ and $A^{\uparrow}$ are the corresponding weighting values for frequency mixing. However, directly predicting $A^{\downarrow}$, $A^{\uparrow} \in \mathbb{R}^{C\times H\times W}$ can be computationally heavy. Thus, we decompose the weighting values prediction into channel-wise and spatial-wise components

$$f'(c,h,w) = A_1^{\downarrow}(c) * A_2^{\downarrow}(h,w) * f^{\downarrow}(c,h,w) + A_1^{\uparrow}(c) * A_2^{\uparrow}(h,w) * f^{\uparrow}(c,h,w), \tag{6}$$

where the channel-wise prediction can be simply implemented by global pooling followed by a convolution layer, and spatial-wise prediction can be implemented by a convolution layer. We use the sigmoid function to regularize the weighting values to be in the range $[0, 1]$. FreqMix can easily integrate into the encoder block of existing models (He et al., 2016; Wang et al., 2023).

4 EXPERIMENTS

4.1 METRIC AND DATASET

Metric. Following previous works (Long et al., 2015; Cheng et al., 2022), we employ standard metrics like mean intersection-over-union (mIoU). For a finer evaluation of boundary segmentation accuracy, inspired by (Cheng et al., 2021), we use mean boundary IoU and accuracy (BIoU, BAcc). Additionally, we introduce FErr, MErr, and DErr from equation 4 to assess distinct hard pixel errors at boundaries: false responses, merging errors, and displacements.

Datasets. For semantic segmentation, we employ two widely-used and challenging datasets: Cityscapes (Cordts et al., 2016) and ADE20K (Zhou et al., 2017). Cityscapes (Cordts et al., 2016) comprises a collection of 5,000 finely annotated images, each with dimensions of 2048×1024 pixels. This dataset is meticulously divided into 2,975 images for the training set, 500 for the validation set, and 1,525 for the testing set. It encompasses 19 distinct semantic categories designed specifically for semantic segmentation tasks. The ADE20K (Zhou et al., 2017), on the other hand, is a notably challenging dataset that encompasses a whopping 150 semantic classes. This extensive dataset features a distribution of 20,210 images for training, 2,000 for validation, and 3,352 for the test set. For low-light instance segmentation, we use the low-light instance segmentation dataset LIS (Chen et al., 2023) to evaluate the proposed method. It provides 8 common classes and consists of 2,230 pairs of low-light and normal-light images, with 70% for training and 30% for testing.

4.2 IMPLEMENT DETAILS

All UPerNet (Xiao et al., 2018) is trained on the Cityscapes dataset using a crop size of 768×768, a batch size of 8, and a total of 80K iterations. We employ stochastic gradient descent (SGD) with a momentum of 0.9 and a weight decay of 5e-4. The initial learning rate is set at 0.01. During training, we adjust the learning rate using the common 'poly' learning rate policy, which reduces the initial learning rate by multiplying $(1 - \frac{\text{iter}}{\text{max\_iter}})^{0.9}$. We apply standard data augmentation techniques, including random horizontal flipping and random resizing within the range of 0.5 to 2. For InternImage (Wang et al., 2023) on ADE20K, we use the same training settings in the paper (Wang et al., 2023). For PointRend (Kirillov et al., 2020) and Mask2Former (Cheng et al., 2022) on the LIS dataset, we adopt the same training settings as described in the paper (Chen et al., 2023).

4.3 ABLATION STUDY

Here, conduct experiments to verify the effectiveness of the proposed De-aliasing filter(DAF) and frequency mixing module (FreqMix). We adopt widely used UPerNet (Xiao et al., 2018) with ResNet-50 (He et al., 2016) as the segmentation model owing to its effectiveness and simplicity.

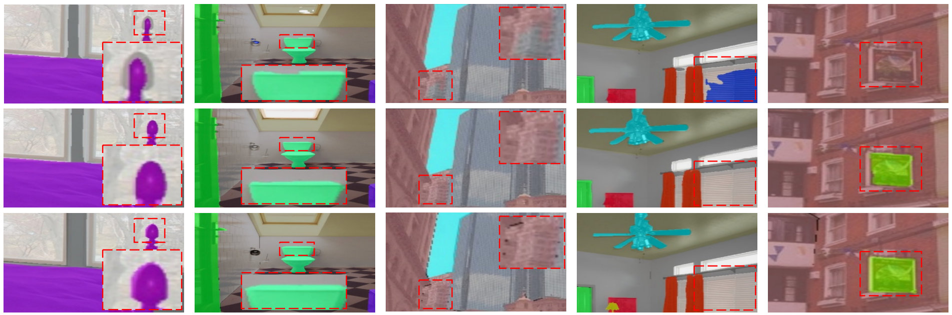

Figure 7: Visualized results on ADE20K. The top, middle, and bottom rows represent vanilla InternImage prediction, prediction of InternImage with our proposed methods, and the ground truth, respectively.

Low-pass cut-off frequency of DAF. Previous work (Grabinski et al., 2022) determines the low-pass cut-off frequency simply by the size of the downsampling layer, *i.e.*, $\frac{1}{4}$. As shown in Table 3. By reducing all frequencies above $\frac{1}{4}$ to prevent aliasing, it improves the vanilla UPerNet by 0.8 mIoU and reduces the displacement error by 0.5, but it also led to an increase in false responses and merging mistakes. By calculating the actual sampling rate with equation 1, we estimate a more accurate equivalent Nyquist frequency of $\frac{\sqrt{2}}{4}$, which achieves the best overall results (mIoU +1.5, BIoU +1.0) among settings where the low-pass cut-off frequency ranged from $\frac{1}{4}$ to $\frac{1}{4} \times 1.6$.

Table 3: Ablation study for low-pass cut-off frequency on the Cityscapes validation set. A higher low-pass cut-off frequency allows more high-frequency components to pass through.

Cut-off frequency	mIoU↑	Boundary		Three type errors			Aliasing Score
		BIoU↑	BAcc↑	FErr↓	MErr↓	DErr↓	
-	77.8	57.4	73.0	25.4	53.3	27.6	9.4
$\frac{1}{4} \times 1.0$	78.6	57.7	73.5	25.8	53.5	27.1	0.0
$\frac{1}{4} \times 1.1$	78.7	58.0	74.0	25.0	**53.2**	26.7	0.0
$\frac{1}{4} \times 1.2$	78.8	58.3	74.1	27.7	53.3	26.7	0.0
$\frac{1}{4} \times 1.3$	79.2	58.1	74.0	24.5	53.3	26.6	0.0
$\frac{1}{4} \times \sqrt{2}$	**79.3**	**58.4**	**74.4**	**24.4**	**53.2**	**26.3**	0.0
$\frac{1}{4} \times 1.5$	78.9	58.1	73.7	26.3	**53.2**	27.0	1.51
$\frac{1}{4} \times 1.6$	79.1	57.8	73.7	25.6	**53.2**	27.0	3.17

Effectiveness of FreqMix. As shown in Table 4, with spatial-wise weighting, it improves the overall performance. Further weighting the frequency for the channel-wise component leads to an additional improvement, totaling +0.6 mIoU while reducing all three types of errors. Meanwhile, it only adds a minimal increase in FLOPs and parameters.

Table 4: Comparison with blur filters on Cityscapes validation set.

Method	mIoU↑	BIoU↑	BAcc↑	FErr↓	MErr↓	DErr↓	#FLOPs	#Params
UPerNet-R50 (Xiao et al., 2018)	77.8	57.4	73.0	25.4	53.3	27.6	297.96G	31.16M
Blur (Zhang, 2019)	78.8	58.0	73.6	26.8	53.4	27.0	298.49G	31.16M
AdaBlur (Zou et al., 2020)	78.9	58.5	73.7	26.7	53.5	26.8	336.56G	32.32M
FLC (Grabinski et al., 2022)	78.6	57.7	73.5	25.8	53.5	27.1	297.96G	31.16M
Ours (DAF)	79.3	58.4	74.4	24.4	53.2	26.3	297.96G	31.16M
Ours (DAF + FreqMix w/o channel-wise)	79.4	58.5	74.5	24.3	53.0	26.2	298.54G	31.22M
Ours (DAF + FreqMix)	**79.7**	**58.8**	**74.9**	**24.0**	**52.8**	**26.1**	298.61G	32.47M

4.4 SEMANTICE SEGMENTATION

Comparison with state-of-the-art blur filters. As shown in Table 4, we compare the proposed DAF with former state-of-the-art blur filters designed to alleviate aliasing, including Blur (Zhang, 2019), AdaBlur (Zou et al., 2020), and FLC (Grabinski et al., 2022). Due to a more precise calculation of the actual sampling rate and the effective removal of the frequency power that leads to aliasing, the proposed DAF achieves the best results. Furthermore, when combined with FreqMix, it leads to further overall improvements, outperforming the best competitor, AdaBlur, by 0.8 mIoU, and achieving the lowest error rates for all three types of hard pixels.

Comparison with state-of-the-art large model. As shown in Tables 5 and 6, we also apply the proposed method with the large-scale model InternImage-L (Wang et al., 2023). It demonstrates an improvement of +0.6/0.4 in mIoU and BIoU on the challenging ADE20K dataset (Zhou et al., 2017) while reducing three types of errors with minor additional FLOPs and parameters. These results highlight the effectiveness of our proposed method even with a large model.

Table 5: Quantitative comparisons of large semantic segmentation models on the ADE20K validation set.

Method	#Params	#FLOPS	mIoU	
			SS	MS
Swin-L (Liu et al., 2021b)	234M	2468G	52.1	53.5
RepLKNet-31L (Ding et al., 2022)	207M	2404G	52.4	52.7
ConvNeXt-L (Liu et al., 2022)	235M	2458G	53.2	53.7
ConvNeXt-XL (Liu et al., 2022)	391M	3335G	53.6	54.0
InternImage-L (Wang et al., 2023)	256M	2526G	53.9	54.1
InternImage-L (Ours)	258M	2529G	**54.5**	**54.6**

Table 6: Evaluation with the large model on the ADE20K validation set.

Method	mIoU↑	BIoU↑	BAcc↑	FErr↓	MErr↓	DErr↓	#FLOPs	#Params
InternImage-L	53.9	42.9	57.8	**13.4**	58.0	42.9	2526.0G	255.6
InternImage-L (Ours)	**54.5**	**43.3**	**58.1**	**13.4**	**57.3**	**42.5**	2528.9G	258.3

Visualized results. As depicted in Figure 7, our proposed method considerably

Table 7: Quantitative comparisons for low-light instance segmentation are conducted on the LIS test set (Chen et al., 2023), utilizing PointRend (Kirillov et al., 2020) and Mask2Former (Cheng et al., 2022) as instance segmentation models, with ResNet-50-FPN (He et al., 2016; Lin et al., 2017a) as the backbone. The training and testing settings align with those outlined in (Chen et al., 2023).

Pipeline	Preprocessing method	Method	AP^{mask}	AP^{box}
Direct	-	PointRend	20.6	23.5
Enhance + Denoise	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	PointRend	26.9	31.2
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	PointRend	27.7	31.9
Integrated Enhance + Denoise	SID (Chen et al., 2018)	PointRend	28.3	31.6
	REDI (Lamba & Mitra, 2021)	PointRend	24.0	27.7
End-to-end (Chen et al., 2023)	-	PointRend	32.8	37.1
End-to-end (**Ours**)	-	PointRend	**34.0 (+1.2)**	**38.2 (+1.1)**
Direct	-	Mask2Former	21.4	22.9
Enhance + Denoise	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	Mask2Former	28.0	30.9
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	Mask2Former	29.3	31.9
Integrated Enhance + Denoise	SID (Chen et al., 2018)	Mask2Former	31.7	33.2
	REDI (Lamba & Mitra, 2021)	Mask2Former	26.7	28.1
End-to-end(Chen et al., 2023)	-	Mask2Former	35.6	37.8
End-to-end (**Ours**)	-	Mask2Former	**36.7 (+1.1)**	**38.8 (+1.0)**

enhances semantic segmentation quality, resulting in a substantial reduction in all three error types, including false responses, merging errors, and displacements. These are consistent with the quantitative results.

4.5 LOW-LIGHT INSTANCE SEGMENTATION

We assess the effectiveness of our proposed method for the challenging task of low-light instance segmentation. The presence of noise in low-light images results in a high aliasing level, which can severely impair performance. As demonstrated in Table 7, our proposed method exhibits a substantial performance improvement (+1.2 AP for PointRend, +1.1 AP for Mask2Former) when compared to the previously established state-of-the-art pipelines described in (Chen et al., 2023).

5 CONCLUSION

In this study, we analyze three types of hard pixels that occur at object boundaries and establish a quantitative link between them and aliasing degradation. Specifically, we propose a method to calculate equivalent sampling rates for estimating the Nyquist frequency and aliasing levels during specific downsampling operations, employing a kernel larger than the stride and involving channel expansion. Moreover, we demonstrate the existence of a positive correlation between pixel-wise segmentation errors and the newly introduced aliasing score. Remarkably, these three types of challenging pixels exhibit distinct and unique patterns within the aliasing score, hinting at potential solutions tailored to scenarios that are sensitive to different types of errors.

These analyses empower us to precisely adjust the aliasing frequencies above the Nyquist frequency for specific downsampling operations. In this regard, we present two concrete solutions. The first solution, the De-Aliasing Filter (DAF), focuses on the modification of the downsampling layer, precisely removing frequencies that lead to aliasing. The second solution, the Frequency Mixing (FreqMix) module, involves a holistic adaptation of the encoder block, aiming to adjust the frequency distribution during feature extraction. Experimental results effectively validate the improved performance in both semantic segmentation and low-light instance segmentation tasks.

The insights gained from this analysis and the measurement tools we have introduced provide a fresh perspective and hold the potential for unlocking numerous research opportunities in this field.

6 LIMITATION

The equivalent sampling rate (ESR) introduced in this study represents our preliminary attempt to calculate the Nyquist frequency for complex downsampling operations that incorporate learnable large kernels and channel expansion. It serves as a heuristic solution for selecting the cutoff frequency. Currently, we only provide an empirical formula for estimation. We believe that there exists substantial potential for a more precise description with better theoretical guarantees. Besides, we have focused on exploring three types of hard pixels that occur at object boundaries in semantic segmentation and their relation to aliasing. However, related tasks such as instance segmentation and panoptic segmentation remain to be investigated.

REFERENCES

Dong Bo, Wang Pichao, and Fan Wang. Afformer: Head-free lightweight semantic segmentation with linear transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1–9, 2023.

Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3291–3300, 2018.

Linwei Chen, Ying Fu, Kaixuan Wei, Dezhi Zheng, and Felix Heide. Instance segmentation in the dark. *International Journal of Computer Vision*, 131(8):2198–2218, 2023.

Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. Boundary iou: Improving object-centric image segmentation evaluation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 15334–15342, 2021.

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1290–1299, 2022.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3213–3223, 2016.

Xueqing Deng, Peng Wang, Xiaochen Lian, and Shawn Newsam. Nightlab: A dual-level architecture with hardness detection for segmentation at night. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 16938–16948, 2022.

Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. Boundary-aware feature propagation for scene segmentation. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 6819–6829, 2019.

Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 13733–13742, 2021.

Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11963–11975, 2022.

Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7890–7899, 2020.

Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, et al. U-net: deep learning for cell counting, detection, and morphometry. *Nature methods*, 16(1):67–70, 2019.

Julia Grabinski, Steffen Jung, Janis Keuper, and Margret Keuper. Frequencylowcut pooling-plug and play against catastrophic overfitting. In *Proceedings of European Conference on Computer Vision*, pp. 36–57, 2022.

Shuhang Gu, Yawei Li, Luc Van Gool, and Radu Timofte. Self-guided network for fast image denoising. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 2511–2520, 2019.

Zhangxuan Gu, Li Niu, Haohua Zhao, and Liqing Zhang. Hard pixel mining for depth privileged semantic segmentation. *IEEE Transactions on Multimedia*, 23:3738–3751, 2020.

Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1780–1789, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Md Tahmid Hossain, Shyh Wei Teng, Guojun Lu, Mohammad Arifur Rahman, and Ferdous Sohel. Anti-aliasing deep image classifiers using novel depth adaptive blurring and activation function. *Neurocomputing*, 536:164–174, 2023.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 17853–17862, 2023.

Shihua Huang, Zhichao Lu, Ran Cheng, and Cheng He. Fapn: Feature-aligned pyramid network for dense image prediction. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 864–873, 2021.

Zhipeng Huang, Zhizheng Zhang, Cuiling Lan, Zheng-Jun Zha, Yan Lu, and Baining Guo. Adaptive frequency filters as efficient global token mixers. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 1–11, 2023.

Yifan Jiang, Xinyu Gong, Ding Liu, Yu Cheng, Chen Fang, Xiaohui Shen, Jianchao Yang, Pan Zhou, and Zhangyang Wang. Enlightengan: Deep light enhancement without paired supervision. *IEEE Transactions on Image Processing*, 30:2340–2349, 2021.

Steffen Jung and Margret Keuper. Spectral distribution aware image generation. In *Association for the Advancement of Artificial Intelligence*, volume 35, pp. 1734–1742, 2021.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *NeurIPS*, 34:852–863, 2021.

Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9799–9808, 2020.

Mohit Lamba and Kaushik Mitra. Restoring extremely dark images in real time. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3487–3497, 2021.

Qiufu Li, Linlin Shen, Sheng Guo, and Zhihui Lai. Wavecnet: Wavelet integrated cnns to suppress aliasing effect for noise-robust image classification. *IEEE Transaction on Image Process.*, 30: 7074–7089, 2021.

Xiangtai Li, Xia Li, Li Zhang, Guangliang Cheng, Jianping Shi, Zhouchen Lin, Shaohua Tan, and Yunhai Tong. Improving semantic segmentation via decoupled body and edge supervision. In *Proceedings of European Conference on Computer Vision*, pp. 435–452. Springer, 2020a.

Xiangtai Li, Ansheng You, Zhen Zhu, Houlong Zhao, Maoke Yang, Kuiyuan Yang, Shaohua Tan, and Yunhai Tong. Semantic flow for fast and accurate scene parsing. In *Proceedings of European Conference on Computer Vision*, pp. 775–793. Springer, 2020b.

Xiaoxiao Li, Ziwei Liu, Ping Luo, Chen Change Loy, and Xiaoou Tang. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3193–3202, 2017.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, 2017a.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pp. 2980–2988, 2017b.

Jiawei Liu, Jing Zhang, Yicong Hong, and Nick Barnes. Learning structure-aware semantic segmentation with image-level supervision. In *International Joint Conference on Neural Networks*, pp. 1–8, 2021a.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 10012–10022, 2021b.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11976–11986, 2022.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 3431–3440, 2015.

Carlos Lopez-Molina, Bernard De Baets, and Humberto Bustince. Quantitative error measures for edge detection. *Pattern recognition*, 46(4):1125–1139, 2013.

Andres Milioto and Cyrill Stachniss. Bonnet: An open-source training and deployment framework for semantic segmentation in robotics using cnns. In *IEEE International Conference on Robotics and Automation*, pp. 7094–7100. IEEE, 2019.

Harry Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644, 1928.

Zequn Qin, Pengyi Zhang, Fei Wu, and Xi Li. Fcanet: Frequency channel attention networks. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 783–792, 2021.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *Proceedings of International Conference on Machine Learning*, pp. 5301–5310, 2019.

Claude E Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 761–769, 2016.

Xin-Yi Tong, Gui-Song Xia, Qikai Lu, Huanfeng Shen, Shengyang Li, Shucheng You, and Liangpei Zhang. Land-cover classification with high-resolution remote sensing images using transferable deep models. *Remote Sensing of Environment*, 237:111322, 2020.

Dongyi Wang, Ayman Haytham, Jessica Pottenburgh, Osamah Saeedi, and Yang Tao. Hard attention net for automatic retinal vessel segmentation. *IEEE Journal of Biomedical and Health Informatics*, 24(12):3384–3396, 2020.

Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 14408–14419, 2023.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of European Conference on Computer Vision*, pp. 418–434, 2018.

Kai Xu, Minghai Qin, Fei Sun, Yuhao Wang, Yen-Kuang Chen, and Fengbo Ren. Learning in the frequency domain. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1740–1749, 2020.

Zhiqin John Xu and Hanxu Zhou. Deep frequency principle towards understanding why deeper learning is faster. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10541–10550, 2021.

Jin Yin, Pengfei Xia, and Jingsong He. Online hard region mining for semantic segmentation. *Neural Processing Letters*, 50:2665–2679, 2019.

Richard Zhang. Making convolutional networks shift-invariant again. In *Proceedings of International Conference on Machine Learning*, pp. 7324–7334, 2019.

Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4106–4115, 2019.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 633–641, 2017.

Lanyun Zhu, Deyi Ji, Shiping Zhu, Weihao Gan, Wei Wu, and Junjie Yan. Learning statistical texture for semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12537–12546, 2021.

Xueyan Zou, Fanyi Xiao, Zhiding Yu, and Yong Jae Lee. Delving deeper into anti-aliasing in convnets. In *Proceedings of the British Machine Vision Conference*, pp. 1–13, 2020.