國立臺灣大學資訊學院資訊網路與多媒體研究所

碩士論文

Graduate Institute of Networking and Multimedia

College of Electrical Engineering and Computer Science

National Taiwan University

Master's Thesis

BEVANet: 雙分支高效視覺注意力網路於即時語義分割

BEVANet: Bilateral Efficient Visual Attention Network for Real-time Semantic Segmentation

黃秉茂

Ping-Mao Huang

指導教授: 莊永裕 博士

Advisor: Yung-Yu Chuang, Ph.D.

中華民國 114 年 2 月

February, 2025

doi:10.6342/NTU202500107

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

雙分支高效視覺注意力網絡於即時語義分割

Bilateral Efficient Visual Attention Network for Real-time semantic segmentation

本論文係 **黃秉茂** （學號 R11944024）在國立臺灣大學資訊網路與多媒體研究所完成之碩士學位論文，於民國 114 年 1 月 10 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Graduate Institute of Networking and Multimedia on 10 January 2025 have examined a Master's Thesis entitled above presented by PING-MAO HUANG (student ID: R11944024) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

莊永裕	葉正聖	吳賦哲
（指導教授 Advisor）		

系（所）主管 Director: 鄭卜壬

doi:10.6342/NTU202500107

Acknowledgements

這篇論文的完成，我想首先感謝我的指導教授莊永裕老師。感謝莊老師在這兩年的碩士學習過程中，給予我無數的建議與支持。在每次的團體會議中，老師總能指出我實驗中的問題與盲點，在很多時候提示我研究方向並即時改正我錯誤的認知。並提到許多我未曾注意到的觀點，使我更清楚如何聚焦於真正重要的部分。透過這些經驗，我學會了如何設計實驗與進行研究。

再來，我要特別感謝亦天還有家緯，感謝你們協助製圖並提供各種呈現上的建議，使我的論文能夠更精確地表達內容。此外，我要感謝 CMLAB 實驗室的所有成員，尤其是 G 組的夥伴們——季嘉、湧致、駱安、力仁、菩提、啟維，以及學長正輝和千祐，還有其他組的朋友們。謝謝你們在這段學習旅程中提供了支持和鼓勵。我們一起研究、共同進步，在學術追求之外也共享生活的喜悅。這種友誼的氛圍，為我的碩士生活增添了許多快樂和動力。感謝你們一直以來的陪伴，讓我能過開心的碩士生涯，也讓我的碩士生活充滿色彩。

最後，我要感謝我的家人們。你們的理解和支持是我學術旅程中最堅強的後盾。感謝你們的支持與鼓勵，使我無需為經濟或其他生活雜事而煩惱，能夠全力專注於學業，並順利完成這篇論文。

雖然這份碩士學位完成的稍久，但我學會了許多新知識，也完成了許多事情。再次感謝每一位在這段旅程中陪伴我的人。

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

摘要

即時語義分割的發展面臨挑戰，在於設計高效的卷積神經網路（Convolutional Neural Network, CNN）或減少視覺轉換器（Vision Transformers, ViT）的計算量。儘管視覺轉換器具長距離依賴性優勢，但運算速度受限。即使大核卷積神經網路提供相似感受野，卻難以適應多尺度特徵與整合全局資訊。為了解決這些問題，我們引入了大核注意力機制（Large Kernel Attention, LKA）提出雙邊高效視覺注意力網路（Bilateral Efficient Visual Attention Network, BEVAN）。高效視覺注意力模組（Efficient Visual Attention, EVA）透過稀疏分解大可分離核注意力（Sparse Decomposed Large Separable Kernel Attentions, SDLSKA），結合區域卷積與條狀卷積與多條拓撲來擴展感受野，捕捉多尺度的全局概念及視覺與結構特徵。而全面核篩選模塊（Comprehensive Kernel Selection, CKS）可動態調整感受野，進一步提升效能。深層大核金字塔池化模組（Deep Large Kernel Pyramid Pooling Module, DLKPPM）結合擴張卷積（Dilated Convolution）與大核注意力機制豐富上下文特徵。雙邊架構（Bilateral Architecture）促進分支間的頻繁訊息交流，而邊界引導注意力融合模塊（Boundary Guided Attention Fusion, BGAF）透過邊界自適應地融合低階空間和高階語義，增強識別模糊邊界的能力。

關鍵字：電腦視覺、即時語義分割、大核注意力、自適應特徵融合

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

Abstract

The development of real-time semantic segmentation faces significant challenges in designing efficient convolutional neural network (CNN) architectures or minimizing the computational costs of vision transformers (ViTs) while maintaining real-time performance. Although ViTs excel at capturing long-range dependencies, their computational speed is often a bottleneck. Large-kernel CNNs offer similar receptive fields but struggle with multi-scale feature adaptation and global context integration. To overcome these limitations, we introduce the Large Kernel Attention mechanism. Our proposed Bilateral Efficient Visual Attention Network (BEVAN) integrates the Efficient Visual Attention (EVA) module, Deep Large Kernel Pyramid Pooling Module (DLKPPM), and Boundary Guided Attention Fusion (BGAF) module. The EVA models expands the receptive field to capture multi-scale contextual information and extracts visual and structural features using Sparse Decomposed Large Separable Kernel Attentions (SDLSKA) by combining regional and strip convolutions with diverse topological structures. The Comprehensive

doi:10.6342/NTU202500107

Kernel Selection (CKS) mechanism dynamically adapts the receptive field to further enhance performance. The Deep Large Kernel Pyramid Pooling Module (DLKPPM) enriches contextual features and extends the receptive field through a combination of dilated convolution and large kernel attention mechanisms, balancing performance and accuracy by refining features and improving semantic concept capture. The bilateral architecture facilitates frequent communication between branches, and the BGAF module uses the guidance of boundary information to adaptively merge low-level spatial features with high-level semantic features, enhancing the network's ability to accurately delineate blurred boundaries while retaining detailed contours and semantic context. Our model achieves a 79.3% mIoU without pretraining, indicating a low dependency on extensive pretraining datasets. After pretraining on ImageNet, the model further attains an 81.0% mIoU, setting a new state-of-the-art benchmark while maintaining real-time efficiency with a processing rate of 32 FPS.

Keywords: Computer Vision, Real-time Semantic Segmentation, Large Kernel Attention, Adaptive Feature Fusion

doi:10.6342/NTU202500107

Contents

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

List of Figures

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

List of Tables

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

Denotation

ViT	Vision Transformers
BEVAN	Bilateral Efficient Visual Attention Network
EVA	Efficient Visual Attention
BGAF	Boundary Guided Attention Fusion
SDLSKA	Sparse Decomposed Large Separable Kernel Attentions
CKS	Comprehensive Kernel Selection
DLKPPM	Deep Large Kernel Pyramid Pooling Module
FCN	Fully Convolutional Networks
CNN	Convolutional Neural Networks
PID	Proportional Integral Derivative
MHSA	Multi-Head Self-Attention
LKA	Large Kernel Attention

doi:10.6342/NTU202500107

TBN	Two-Branch Networks
VAN	Visual Attention Network
feat	feature
OHEM	Online Hard Example Mining
Conv	Convolutional Neural Networks
BSConv	Blueprint Separable Convolutions
BN	Batch Normalization
UP	UPsampling operator
GAP	Global Average Pooling
$GAP_{i\times i}$	Global Average Pooling with resulting size i × i
AP	Average Pooling
$AP_{i\times i,sj}$	Average Pooling with with kernel size i and stride j
$Conv_{i\times i}$	Convolutional Neural Networks with i × i kernel
$DConv_{i\times j,rk}$	Dilated Convolutional Neural Networks using an i × j kernel with a dilation rate of k
σ	Balanced Weight

doi:10.6342/NTU202500107

Chapter 1 Introduction

Semantic segmentation is a fundamental task in computer vision that involves assigning a class label to each pixel in an image, facilitating detailed scene understanding. It requires precise object boundary detection, semantic context comprehension, and object completeness, making it a dense prediction task. The objective is not only to improve performance, but also to enhance efficiency. In practical and real-world scenarios, achieving real-time performance is crucial, necessitating a frame rate of 30 FPS or higher. Additionally, maintaining low computational complexity is essential to ensure fast and efficient processing. It plays a critical role in applications such as autonomous driving [16, 25], medical imaging [1], and robots [50, 53], where precision at the pixel level is essential.

Since the introduction of Fully Convolutional Networks (FCN) [43], which established end-to-end dense prediction, the field has seen rapid advancements. Architectures like UNet [51] and its successors improved segmentation performance by incorporating skip connections and encoder-decoder structures, effectively balancing global context with fine details. PSPNet [76] enhanced results using pyramid pooling. In addition, numerous other notable models have also been introduced [2, 59, 77]. Especially, backbones utilizing dilated convolutions [5–8, 69] combined with context extraction modules have become a widely adopted standard in various semantic segmentation methods. However, the computational complexity of these models often limits their applicability in real-time

doi:10.6342/NTU202500107

scenarios, making them unsuitable for latency-sensitive applications.

Real-time semantic segmentation seeks to overcome this limitation by designing efficient architectures that balance speed and accuracy. These applications require models that are both accurate and computationally efficient, with the added constraint of processing at speeds faster than 30 FPS. Early efforts, such as DFANet [33] and MobileNets [26, 54] employed lightweight depth-wise separable convolutions, ENet [48] utilized a lightweight decoder and downsampled feature maps during the early stages, ICNet [75] processed small-sized inputs through a complex and deep pathway to capture high-level semantics. More recent innovations, like Two-Branch Network (TBN) designs exemplified by BiSeNets [56, 70, 71], STDC [15], and DDRNet [23], have achieved promissing performance by effectively combining low-level spatial features with high-level semantic context. Furthermore, PIDNet [67] applies the principles of Proportional-Integral-Derivative (PID) controllers, incorporating three branches to process detailed, contextual, and boundary information, achieving state-of-the-art performance. Despite these advancements, achieving real-time performance often involves trade-offs, particularly in handling complex scenarios with intricate boundaries or small objects, as well as insufficient receptive fields for capturing contours.

Over the past decade, Convolutional Neural Network (CNN) architectures and optimization techniques have evolved rapidly, achieving significant advancements in tasks such as image classification, object detection, and semantic segmentation. While CNNs augmented with attention mechanisms have proven effective, the emergence of self-attention-based networks, such as Vision Transformers (ViT) [14] and their variants [38, 41, 61, 64], has redefined the field. The superior performance of ViTs is primarily attributed to their ability to model long-range dependencies using Multi-Head Self-Attention (MHSA),

doi:10.6342/NTU202500107

along with their better scalability. However, this performance comes at the cost of a quadratic increase in computational and memory requirements, particularly for high-resolution inputs, limiting their practicality in real-time and resource-constrained scenarios. SegFormer [65], RTFormer [58], and SeaFormer [57] employ efficient attention mechanisms. However, the approaches incorporating dense fusion modules between branches to enhance the semantic richness of extracted features are computationally intensive.

To bridge this gap, Large Kernel Attention (LKA) [18, 19, 32, 40] has emerged as a promising alternative, combining the strengths of convolution and attention mechanisms. LKA captures long-range dependencies and global context more effectively by leveraging large receptive fields. Recent approaches like RepLKNet [13] demonstrate the benefits of Large Kernel Attention (LKA), integrating convolution and attention mechanisms to capture global context effectively. SLaK [40] expanded kernel sizes to 51 by replacing a large kernel with two long parallel kernels and a small kernel. VAN [19] and LSKA [32] further optimized this with dilated and strip convolution, reducing computational demands. LSKNet [37] introduced the selective kernel concept from SKNet [34]. However, these approaches still lack multi-scale feature integration and receptive field adjustment.

Previous methods on real-time semantic segmentation have struggled with accurately handling contour details and effectively capturing semantic context. They also face inefficiencies in fusing low-level and high-level features. Our approach addresses these challenges through a novel framework that enhances feature representation and segmentation performance. We introduce the LKA mechanism into our Bilateral Efficient Visual Attention Network (BEVAN) in real-time semantic segmentation, such as capturing contour details, semantic context, and fusing features at different levels. Our design features the Efficient Visual Attention (EVA) block with Sparse Decomposed Large Separable Kernel

doi:10.6342/NTU202500107

Attention (SDLSKA) to expand the receptive field, capture multi-scale semantic context, and dynamically combine global and local information. We introduce the Comprehensive Kernel Selection (CKS) mechanism, which integrates features from both small and large kernels using dynamic channel and spatial attention. The EVA block enhances both visual and structural feature extractions through region and strip convolutions with diverse topologies. Additionally, we propose the Deep Large Kernel Pyramid Pooling Module (DLKPPM) to enhance contextual features and mitigate information loss often caused by pooling and striding in traditional methods. DLKPPM integrates a large kernel attention mechanism with dilated convolution to expand the receptive field, refine features, and improve semantic representation. We also develop a bilateral architecture that facilitates continuous communication between two branches and the Boundary Guided Attention Fusion (BGAF) module that adapts semantic and detail fusion with boundary information. The interactions of these branches improve the representation of features by integrating various features, improving segmentation accuracy and preserve detailed contours. As shown in Fig. 1.1, BEVAN offers a robust and efficient framework that achieves state-of-the-art performance in real-time semantic segmentation scenarios by effectively balancing accuracy and computational efficiency. Our main contributions can be summarized as follows:

- **Efficient Attention Mechanisms.** We leverage large kernel attention to design the EVA block, SDLSKA, CKS, and DLKPPM modules for dynamically expanding and adjusting receptive fields, enhancing feature representation, capturing semantic concepts, and refining details.
- **Branch Interaction.** Frequent communication between high- and low-level branches through the bilateral architecture and the BGAF module enhances semantic concepts

doi:10.6342/NTU202500107

and detail contour by sharing information, enabling adaptive feature fusion.

- **Performance.** BEVAN balances inference speed and accuracy better compared to existing models. It achieves real-time segmentation over 30 FPS with 81.0% mIoU in Cityscapes after pre-training on ImageNet and maintains 79.3% mIoU without pre-training, showing less dependency on large pre-training datasets.

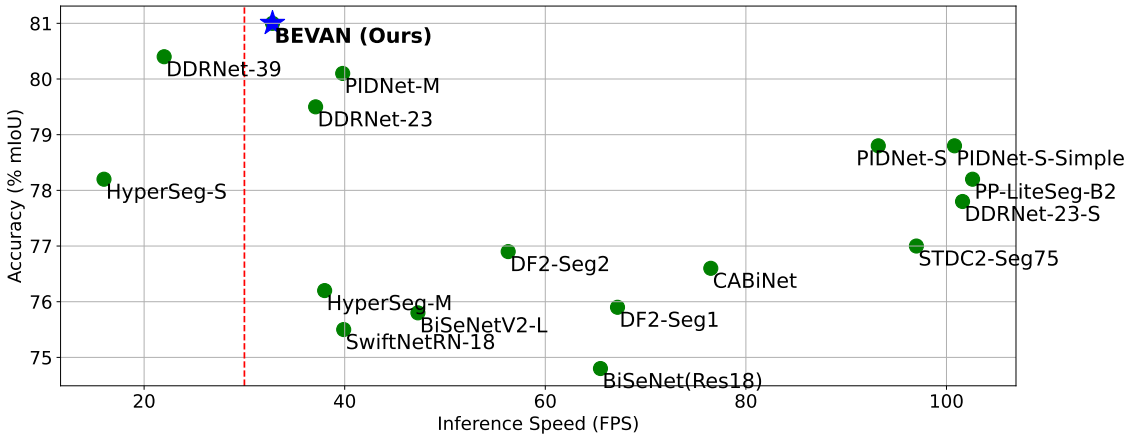

Figure 1.1: **Performance of real-time models on the Cityscapes [11] validation set**, with our model in blue and others in green.

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

Chapter 2 Related Work

2.1 Generic Semantic Segmentation

Traditional segmentation algorithms, e.g., threshold selection, super-pixel, utilized the hand-crafted features to assign pixel-level labels in images. With the development of convolution neural networks, methods based on FCN [43] achieved impressive performance on various benchmarks. The DeepLab series [6–8] adopted an atrous spatial pyramid pooling module to capture multi-scale context. The SegNet [2] utilized the encoder-decoder structure to recover the high-resolution feature maps. The PSPNet [76] devised a pyramid pooling to capture both local and global context information. HRNet [59] emphasizes deep high-resolution representations and demonstrates greater efficiency compared to dilation-based backbones. It utilizes multiple paths and bilateral connections to learn and fuse features across different scales. Their structure can simultaneously learn the low-level details and high-level semantics. However, most approaches require large computation costs due to the high-resolution feature and the complicated network connections. In this paper, we propose an efficient and effective architecture which achieves a good trade-off between speed and accuracy.

doi:10.6342/NTU202500107

2.2 Real-time Semantic Segmentation

Achieving a balance between inference speed and accuracy in semantic segmentation has been a critical driver of innovation in network architectures. Researchers have continuously sought to design models that not only deliver high precision in pixel-level classification but also operate efficiently enough for real-time applications. This dual objective has led to the development of various techniques, including lightweight backbones, multi-scale feature integration, and advanced attention mechanisms. These innovations aim to optimize computational resources while maintaining or improving segmentation performance, addressing the growing demand for high-speed, accurate models in practical scenarios.

SwiftNet [45] uses a dual-input approach, combining low-resolution semantic input and high-resolution detailed input, processed by a lightweight decoder. DFANet [33] adapts Xception [10] with a lightweight backbone using depthwise separable convolutions and reduces input size for faster inference. ICNet [75] speeds up processing through a multi-resolution cascade, while ShuffleSeg [17] reduces computational costs with ShuffleNet's [74] channel shuffling and group convolution. However, encoder-decoder models still face latency due to deep sequential processing, and depthwise separable convolutions aren' t fully optimized on GPUs, making traditional convolutions faster.

Transformer-based methods, such as TopFormer [73], RTFormer [58], and SeaFormer [57], present efficient alternatives for real-time semantic segmentation but encounter several challenges. TopFormer [73] operates at a 1/64 scale, which significantly compromises accuracy, limiting its effectiveness in detailed segmentation tasks. RTFormer [58] and SeaFormer [57] attempt to mitigate the computational overhead of the attention mecha-

doi:10.6342/NTU202500107

nism through various strategies aimed at reducing the attention computation. However, the design of frequent interactions between branches also increases computational demands. Despite their efforts to retain the transformative power of attention mechanisms while cutting down on computation, these strategies often involve parameter reduction, which ultimately weakens the models' performance. This trade-off results in models that are efficient but fail to maintain the high level of accuracy typically expected from transformer-based approaches, highlighting the need for more balanced solutions that can deliver both efficiency and robust performance.

BiSeNets [56, 70, 71] addresses this by introducing a two-branch network (TBN) for context and detailed feature processing, fused through a feature fusion module for better boundary contour and small object recognition. STDC [15], remove the spatial branch and add a detailed guidance module . DDRNet [23] introduces bilateral connections for improved information exchange, optimizing branch sharing for real-time performance. However, direct fusion of detailed semantics and low-frequency context can blur boundaries and obscure small objects. PIDNet [67] utilizes PID controllers across three branches to handle detailed, contextual, and boundary information. It introduces the Pag module for feature fusion with balanced weights and similarity, and the Bag block for fusing branch features under guided supervision. Additionally, it generates pseudo boundary ground truth to supervise the training of the boundary branch. These innovations enable PIDNet [67] to achieve state-of-the-art performance in real-time segmentation.

However, we believe that PIDNet's [67] design is somewhat redundant and overly complex, with multiple branches leading to higher computational costs. Additionally, it lacks a large receptive field, which is crucial for capturing broader semantic context. In response, our work introduces a novel architecture with efficient modules designed to

doi:10.6342/NTU202500107

streamline feature integration, reduce latency, and enhance the preservation of details and semantic information. By incorporating large kernel attention, our approach significantly improves real-time semantic segmentation performance while maintaining efficiency.

2.3 Large Kernel Attention

Over the past decade, CNN [46] architectures have advanced significantly, with most models focusing on 3×3 kernels for computational efficiency. Notable exceptions, such as AlexNet [30] and Inception, experimented with various kernel sizes. However, attempts to scale up kernels, like in LR-Net [27] with 7×7 kernels, encountered performance saturation due to optimization difficulties.

Transformer-based models such as Vision Transformer (ViT) [14], Swin Transformer [41], and Pyramid Transformer [60] have become prominent in computer vision due to their wide receptive fields. However, research shows that well-designed convolutional architectures with large kernels can be just as competitive. Models like SegNeXt [18] and Conv2Former [24] emphasize the importance of large kernel convolutions in improving contextual representation within convolutional features.

Recent advancements have incorporated transformer-inspired concepts, focusing on large receptive fields for enhanced performance. ConvNeXt [42, 62] utilized 7×7 depth-wise convolutions and applied key transformer principles, such as optimized training strategies and adjusted compute ratios. RepLKNet [13] further expanded this approach by utilizing 31×31 kernels through re-parameterization, emphasizing the use of depth-wise convolution to reduce the computational load associated with large kernels. Re-parameterization further aids in cutting down the number of parameters, leading to in-

doi:10.6342/NTU202500107

creased processing speed. Moreover, it demonstrates that large kernels often outperform smaller ones, even with small feature sizes. The whole model results in superior performance compared to models like the Swin Transformer [41]. Additionally, RepLKNet [13] achieves better outcomes in tasks such as ImageNet[52] classification with little higher computational demands.

SLaK [40] expanded kernel sizes to 51×51 by replacing a single large square kernel with two long rectangular parallel kernels and a small square kernel. This approach involves decomposing large kernels into rectangular shapes (e.g., 51×5, 5×51, and 5×5) and fusing them, simplifying large kernel computation and significantly reducing the number of trainable parameters through sparse grouping. The Visual Attention Network (VAN) [19] advanced this concept by combining depthwise, dilated, and pointwise convolutions to form Large Kernel Attention (LKA), with dilated convolutions constructing large kernels using fewer parameters. This method leverages 2D structural information with depthwise and dilated convolutions, demonstrating lower computational demands than self-attention. It adaptively handles spatial and channel dimensions, making it a powerful approach by using features as needed. VAN [19] strikes a balance between capturing long-range dependencies, spatial adaptability, and computational efficiency. LSKA [32] further improved VAN's [19] speed by introducing strip convolutions, enhancing the efficiency of large kernel attention computations.

Despite these advancements, existing models still fall short in terms of efficiency and speed, and their receptive fields lack sufficient adaptability. To address these issues, we propose the Efficient Visual Attention (EVA) Block. We refine Large Kernel Attention (LKA) by introducing Sparse Decomposed Large Kernel Attention (SDLSKA), enhancing its adaptability and efficiency in forming large kernels. Additionally, our Comprehensive

doi:10.6342/NTU202500107

Kernel Selection (CKS) modules capture multi-scale features and dynamically integrate high- and low-level information. This approach improves efficiency while preserving high performance in tasks like semantic segmentation. Our method sets a new benchmark, achieving high accuracy while maintaining low computational demands.

2.4 Feature Fusion

The attention mechanism effectively enhances neural representations through channel and spatial attention. Channel attention methods like SE block [29] reweight feature channels using global averages, while spatial attention modules such as GENet [28] and GCNet [4] improve contextual modeling using spatial masks. Hybrid approaches like BAM [47] and CBAM [63] integrate both types for comprehensive attention.

In addition to attention mechanisms, dynamic kernel selection enables adaptive context modeling. Methods like CondConv [68] and Dynamic Convolution [9] aggregate features from multiple convolution kernels, while SKNet [34] and ResNeSt [72] utilize multi-branch convolutional designs with selective fusion. SCNet [39] further enhances this by integrating spatial attention, and Deformable Convnets [12, 78] introduce flexible kernel shape adjustments. LSKNet [37] brings the SKNet [34] concept into the Visual Attention Network (VAN) [19], replacing channel-wise selection with spatial-wise fusion.

Current methods treat channel-wise and spatial-wise feature fusion independently, overlooking their interdependence, which is crucial for feature integration across different kernel scales. Our model requires the simultaneous integration of more feature sets, demanding a fusion mechanism that adapts to both spatial and channel-wise selection. While SKNet [34] and LSKNet [37] were designed for two kernel features, their separate

doi:10.6342/NTU202500107

handling of these dimensions becomes inadequate as more features are involved, complicating the fusion process. To address this, our Selective Kernel methods enhance by integrating spatial and channel attention. It captures the complex interplay between these dimensions, enabling a more holistic and adaptive fusion of multi-scale features. This approach improves feature representations and performance, ensuring the fusion process leverages the strengths of both dimensions and advancing kernel fusion techniques.

2.5 Pyramid Pooling Module

In semantic segmentation, capturing richer contextual information is vital for enhancing performance. Following the introduction of the Pyramid Pooling Module (PPM) in PSPNet [76], which effectively captures both local and global context by concatenating multi-scale pooling maps, several other pyramid pooling modules have been developed to further improve this process. Atrous Spatial Pyramid Pooling (ASPP) [6, 7, 69] utilizes parallel atrous convolutions with varying rates to capture multi-scale contexts, while Deep Aggregation Pyramid Pooling Module (DAPPM) [23] improves context embedding by combining kernels of different depths and sizes. To address the limitations of DAPPM [23], the Parallel Aggregation Pyramid Pooling Module (PAPPM) [67] was introduced. It further adapted to Parallel structure offering faster inference at the cost of some segmentation accuracy. The Simple Pyramid Pooling Module (SPPM) [49] further simplifies PAPPM [67] by replacing average pooling with global average pooling, increasing efficiency but again compromising accuracy. We propose the Deep Large Kernel Pyramid Pooling Module (DLKPPM), which integrates a large kernel attention mechanism with LSKA [32] and replaces traditional convolution with dilated convolution. This expansion of the receptive field helps refine features and capture semantic concepts more effectively.

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

Chapter 3 Methodology

We propose the Efficient Visual Attention (EVA) module, utilizing Sparse Decomposed Large Separable Kernel Attentions (SDLSKA) and Comprehensive Kernel Selection (CKS) to adaptively enlarge the receptive field. The Deep Large Kernel Pyramid Pooling Module (DLKPPM) leverages large kernels for contextual enrichment. Additionally, the Bilateral Architecture (BA) and Boundary Guided Attention Fusion (BGAF) facilitate feature interaction across two branches.

3.1 Bilateral Architecture

PIDNet [67] leverages Proportional-Integral-Derivative (PID) controller principles, utilizing three branches to process detailed, contextual, and boundary information. However, the three-branch design is inefficient and wastes time on many inter-branch interactions. To improve efficiency, we adapt the architecture by combining the detail and boundary branches into a single low-level branch and removing redundant components, streamlining the model for better performance. As Figure 3.1 shows, we propose a bilateral architecture consisting of two branches.

The high-level branch aggregates contextual information locally and globally, capturing long-range dependencies and providing rich semantic features to the low-level branch.

doi:10.6342/NTU202500107

This interaction helps refine detailed features while the high-level branch continues to extract and compress features for context representation. On the other hand, the low-level branch focuses on preserving detailed information within high-resolution feature maps and extracting high-frequency features to accurately predict boundary regions. By maintaining a resolution of 1/8 of the original size, it avoids excessive compression, ensuring the retention of fine contour details and boundary information. This architecture facilitates continuous interaction between the high-level and low-level feature branches, enhancing both semantic understanding and object boundary precision.

To integrate these branches, we employ the Boundary Guided Adaptive Fusion (BGAF) block, which uses boundary information to guide the fusion of detailed and semantic features, achieving a balanced and precise feature representation and ensuring optimal feature combination for improved predictions.

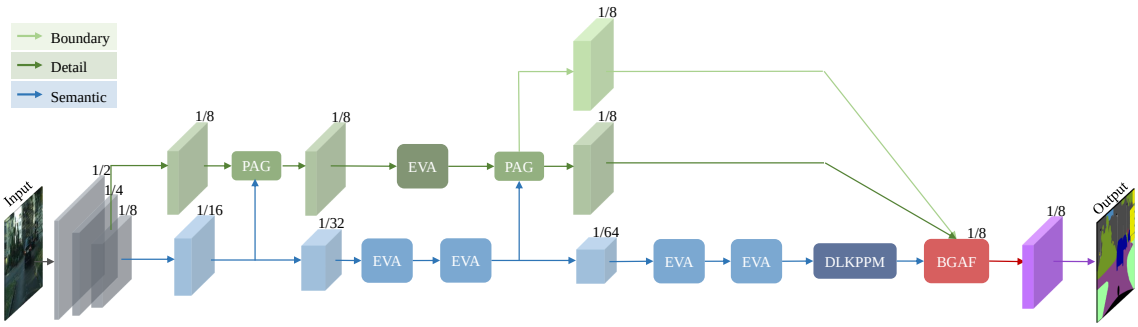

Figure 3.1: **The overall structure of the BEVAN.**

3.2 Efficient Visual Attention Block

Our EVA Block is inspired by the robust block design in VAN [19] and LSKA [32], demonstrated in Fig. 3.2(a). The overall architecture of the EVA Block consists of two main components: the Large Kernel Attention (LKA) and the Convolution Feed-forward Network (CFFN). By introducing Large Kernel Attention and Selective Kernel mecha-

doi:10.6342/NTU202500107

nisms, we enhance the block's adaptability, strength, and ability to process multi-scale features effectively.

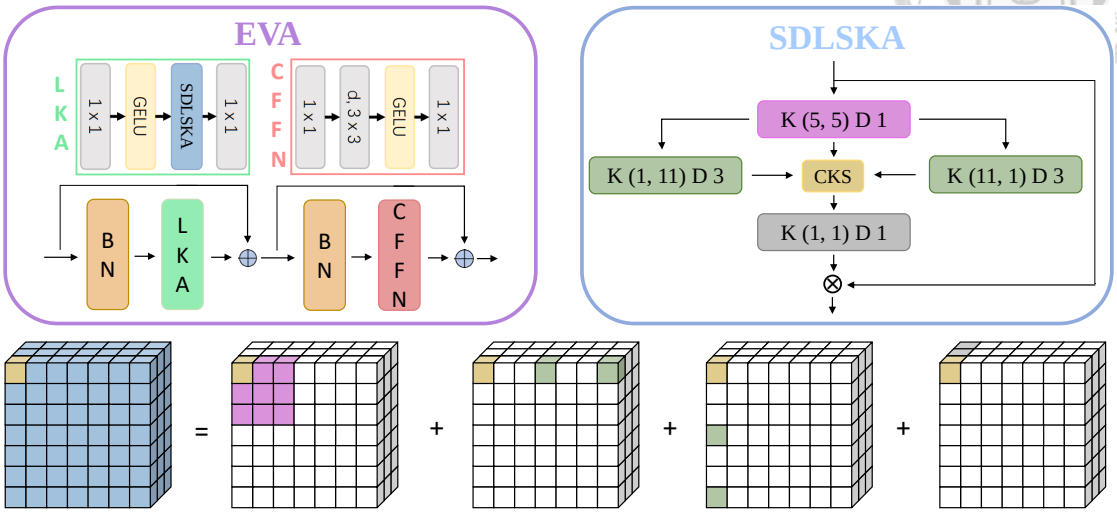

Figure 3.2: **The structure of (a) Efficient Visual Attention (EVA) block and (b) Sparse Decompose Large Separable Kernel Attentions (SDLSKA) module.**

The LKA sub-block captures long-range dependencies. By utilizing Sparse Decomposed Large Separable Kernel Attentions (SDLSKA), it enhances the extraction of accurate semantic information by leveraging a large receptive field. Incorporating a broader contextual view and extensive receptive field improves low-level features, allowing for better preservation of contours and more detailed information. The block also uses a Comprehensive Kernel Selection (CKS) mechanism to combine features from kernels of different shapes, enabling the fused features to represent information more precisely and adaptively. This combination of large receptive field capture and adaptive feature fusion makes the EVA Block an effective and efficient module for various tasks requiring precise feature representation.

The CFFN sub-block refines and integrates features, ensuring that the output is well-balanced and informative. Facilitates channel mixing and feature refinement. This sub-block comprises a fully connected layer, a depthwise convolution, a GELU [22] activation, and a second fully connected layer, enhancing the representation and flow of information

doi:10.6342/NTU202500107

within the network.

3.2.1 Sparse Decompose Large Separable Kernel Attentions

We integrate the strengths of LSKA [32] and SLaK [40] to construct an efficient Large Kernel Attention block. The SDLSKA module is designed to expand the receptive field to effectively capture semantic information and refine details. The structure is depicted in Fig. 3.2(b). Drawing from SLaK [40], we simplify large kernel computation through sparse grouping by decomposing them into a smaller convolution and two strip dilation kernels, then adaptively fusing them using CKS module. The smaller convolution helps focus on specific areas, while the two strip dilation kernels refine the focus, with low computation. Additionally, inspired by LSKA [32], we combine strip convolutions with depthwise, pointwise, and dilated convolutions to capture large kernel features efficiently. This approach reduces parameters while leveraging 2D structural information, resulting in better computational efficiency. It also adapts effectively to spatial and channel dimensions to capture long-range dependencies.

In the Large Kernel Attention sub-block within our EVA Block, features are first passed through a 1×1 point-wise convolution for channel interaction, followed by GELU [22] activation layers to introduce non-linearity. The features then enter the Sparse Decompose Large Separable Kernel Attention module, which refines them as follows:

The features are initially processed through a 5×5 standard convolution to extract small kernel features.

$$small\ kernel\ feat = Conv_{5\times5}(feat)$$

doi:10.6342/NTU202500107

They are then passed through two directional strip convolutions (1×11 and 11×1) with dilation rate 3 to form large kernel features in horizontal and vertical orientations.

$$\text{large h kernel feat} = \text{DConv}_{11\times1,r3}(\text{feat})$$

$$\text{large v kernel feat} = \text{DConv}_{1\times11,r3}(\text{feat})$$

The small and large kernel features are fused using the Comprehensive Kernel Selection (CKS) mechanism, which adaptively combines multi-scale features.

selected feat = CKS(small kernel feat, large h kernel feat, large v kernel feat)

The resulting features are further refined using a 1×1 point-wise convolution for channel interaction and multiplied with the features prior to entering the Sparse Decompose Large Separable Kernel Attention module.

$$out\ feat = feat \otimes Conv_{1\times1}(selected\ feat)$$

This design achieves a theoretical receptive field of 35, enabling precise semantic information capture while maintaining computational efficiency. The combination of multi-scale processing and adaptive feature integration ensures robust performance across a range of tasks.

3.2.2 Comprehensive Kernel Selection

The Comprehensive Kernel Selection (CKS) module in SDLSKA dynamically adjusts the receptive field and fuses multi-scale features by jointly considering channel-wise

doi:10.6342/NTU202500107

and spatial-wise dependencies, unlike SKNet[34] and LSKNet[37], which treat these dimensions separately. This integrated approach is crucial for effective feature fusion, as it captures the interdependence between spatial and channel dimensions, ensuring a more holistic representation. As illustrated in Fig. 3.2(c), our module efficiently manages complex multi-scale fusion across diverse kernel shapes, such as 5×5, 11×1 (dilation=3), and 1×11 (dilation=3), enabling flexible adaptation to feature characteristics. While SKNet introduced channel-wise kernel selection and LSKNet extended it to spatial-wise fusion, both approaches become insufficient as the number of kernel scales increases. Our method overcomes this limitation by simultaneously integrating multiple feature sets, leading to a more adaptable and expressive representation that significantly enhances feature extraction and fusion efficiency.

It computes the weights across both channels and spatial dimensions to make the receptive field adjustments. We feed three features after passing different scales of kernel to get their corresponding weight, and element-wise addition of the feature multiplies with their weights.

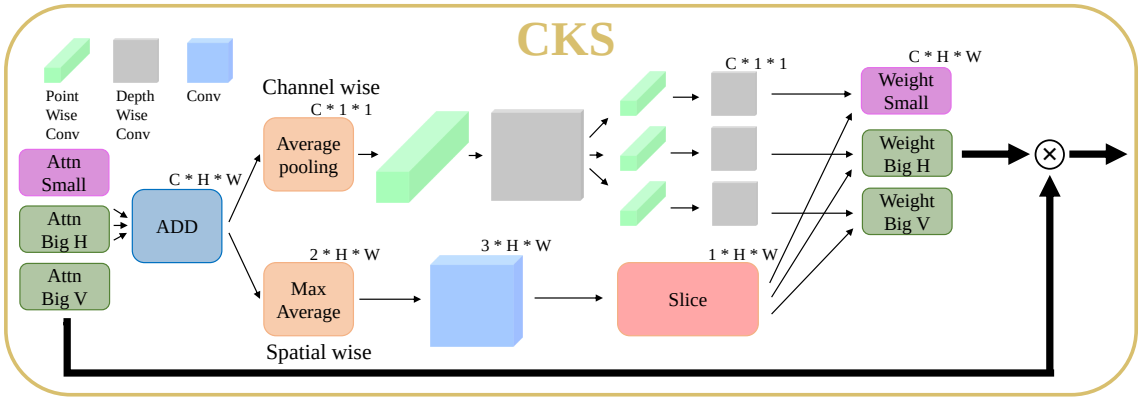

Figure 3.3: **The structure of Comprehensive Kernel Selection (CKS) module.**

As Figure 3.3 shows, we begin by summing the three features, then use pooling to extract the channel-wise feature and aggregation to obtain the spatial-wise feature. Next, we multiply the spatial and channel weights to compute the adjustment weight, which adap-

doi:10.6342/NTU202500107

tively adjusts the receptive field. For channel-wise selection, we apply average pooling, and for spatial-wise selection, we use both max and average pooling. This results in two-dimensional features. In the spatial-wise branch, a convolution operation generates three channels of features, which are then divided into three spatial weights. In the channel-wise branch, we introduce Blueprint Separable Convolutions (BSConv) [20], which consist of a point-wise convolution followed by a depthwise convolution to refine the features. We first pass the features through a BSConv [20] to refine them, and then use three BSConv [20] paths to derive the weight on the channel dimension.

3.3 Deep Large Kernel Pyramid Pooling Module

The DAPPM [23] enhances context embedding by combining kernels of different depths and sizes, but it suffers from large strides or pooling that can overlook finer spatial information. To address this, we propose the Deep Large Kernel Pyramid Pooling Module (DLKPPM), which maintains the hierarchical-residual structure of different scales while complementing the large receptive field. Additionally, we incorporate a large kernel attention mechanism with LSKA [32] to expand the receptive field to 23. For better contextual fusion, we replace traditional convolution with dilated convolution for small kernel feature maps, while retaining standard convolution for large kernel and pooling layers. This approach refines features, captures semantic concepts more effectively, and achieves an optimal balance between performance and accuracy. Considering an input x, each output y_i at different scales can be expressed as:

doi:10.6342/NTU202500107

$$
y_i = \begin{cases}
\text{Conv}_{1\times1}(x) & i = 1, \\
\text{DConv}_{3\times3,r2}\left(\text{UP}\left(\text{Conv}_{1\times1}\left(\text{AP}_{5\times5,s2}\right)\right) + y_{i-1}\right) & i = 2, \\
\text{DConv}_{3\times3,r2}\left(\text{UP}\left(\text{Conv}_{1\times1}\left(\text{AP}_{9\times9,s4}\right)\right) + y_{i-1}\right) & i = 3, \\
\text{Conv}_{3\times3}\left(y_1 + \text{LSKA}(\text{BN}(y_1)) + y_{i-1}\right) & i = 4, \\
\text{Conv}_{3\times3}\left(\text{UP}\left(\text{Conv}_{1\times1}(\text{GAP}_{2\times2})\right) + y_{i-1}\right) & i = 5, \\
\text{Conv}_{3\times3}\left(\text{UP}\left(\text{Conv}_{1\times1}(\text{GAP}_{1\times1})\right) + y_{i-1}\right) & i = 6.
\end{cases}
$$

where $\text{Conv}_{1\times1}$ is a point-wise 1×1 traditional convolution, $\text{Conv}_{3\times3}$ is a 3×3 convolution, $\text{DConv}_{3\times3,r2}$ means a 3×3 dilated convolution with dilation rate 2, UP represents the upsampling operation, $\text{AP}_{i\times i,s_j}$ denotes the average pooling layer with kernel size i and stride j, $\text{GAP}_{i\times i}$ denotes the global average pooling with the resulting size, BN denotes batch normalization, and LSKA stands for the large kernel separable attention block [32]. Finally, all feature representations are concatenated and reduced the channels using a 1×1 point-wise convolution. Additionally, a 1×1 projection shortcut is introduced.

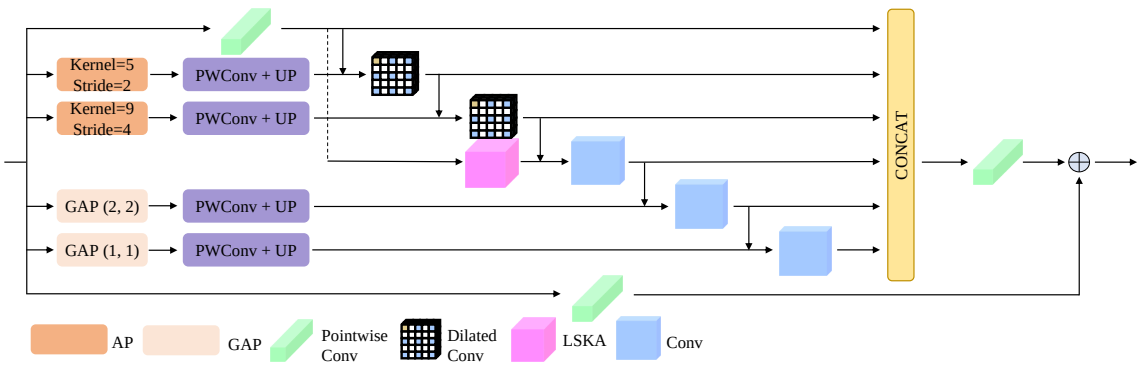

Figure 3.4: **The structure of Deep Large Kernel Pyramid Pooling Module (DLKPPM) module.**

doi:10.6342/NTU202500107

3.4 Boundary Guided Adaptive Fusion

The Boundary Guided Adaptive Fusion (BGAF) module is an efficient multi-branch aggregation framework designed to balance contextual and spatial features by the guidance of the boundary information for more precise predictions. Since simple weighted summation can degrade feature quality, BGAF employs a shortcut connection to preserve critical feature details, ensuring that the fusion process maintains both semantic richness and spatial precision. By dynamically adjusting feature contributions based on boundary significance, it effectively mitigates the limitations of the semantic branch's low spatial accuracy and the detail branch's shallow semantic representation. This results in a refined fusion mechanism that enhances object boundary detection and captures fine-grained structures with higher accuracy. As depicted in Fig. 3.5, BGAF ensures a seamless integration of high-level contextual features and low-level spatial details, integrating semantic understanding with precise contour details based on boundary significance, allowing adaptive weighting that enhances feature expressiveness while maintaining structural integrity.

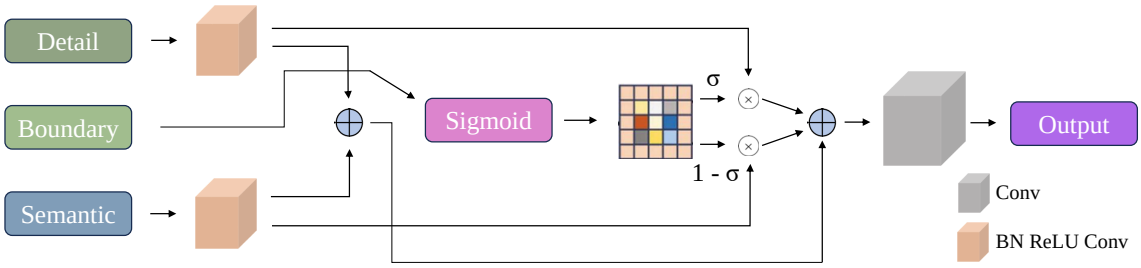

Figure 3.5: **The structure of Boundary Guided Adaptive Fusion (BGAF) module.**

Semantic and detail features are processed using Batch Normalization (BN), ReLU, and a convolutional layer to adjust and refine their representations:

$$\text{FeatS} = \text{Conv}(\text{ReLU}(\text{BN}(\text{FeatS})))$$

$$\text{FeatD} = \text{Conv}(\text{ReLU}(\text{BN}(\text{FeatD})))$$

doi:10.6342/NTU202500107

The boundary feature is processed through a Sigmoid activation function to compute the balancing weight σ, which regulates the contribution of each branch during feature fusion:

$$\sigma = \text{Sigmoid}(\text{FeatB})$$

The semantic and detail features are adaptively merged using the calculated balancing weight σ, where the detail feature is weighted by σ and the semantic feature by 1−σ. This approach emphasizes detailed features around boundaries while preserving contextual information elsewhere.

$$balanced\ feat = FeatD \otimes \sigma + FeatS \otimes (1-\sigma)$$

A 1×1 projection shortcut is formed by element-wise addition of the refined semantic and detail features,

$$\text{shortcut} = \text{FeatS} + \text{FeatD}$$

The balanced feature and shortcut are added element-wise and passed through a final convolutional layer for the output:

$$\text{output} = \text{Conv}(\text{balanced feat} + \text{shortcut})$$

By dynamically adjusting the importance of semantic contextual and spatial detail features based on boundary significance, BGAF ensures precise segmentation, especially in challenging areas such as object boundaries and small structures.

doi:10.6342/NTU202500107

Chapter 4 Experiments

4.1 Dataset

4.1.1 Cityscapes

Cityscapes [11] is a benchmark dataset for high-quality urban scene parsing, consisting of 5,000 finely annotated images captured from a car perspective across various cities. These images are split into 2,975 for training, 500 for validation, and 1,525 for testing. With a high resolution of 2048×1024. Cityscapes [11] presents a significant challenge for real-time semantic segmentation models. The dataset includes 19 annotated classes commonly used for evaluation in semantic segmentation tasks. Only the finely annotated images are used in experiments to ensure fair comparisons with state-of-the-art models.

4.1.2 Camvid

CamVid [3] consists of 701 driving scene images, divided into 367 for training, 101 for validation, and 233 for testing. Each image has a 960×720 resolution, with 32 annotated categories. For fair comparison with previous studies, 11 classes are selected for evaluation.

doi:10.6342/NTU202500107

4.2 Experiment Settings

4.2.1 Pretraining

The models are pre-trained on ImageNet [52] using a standard data augmentation strategy, including random cropping to 224×224 and horizontal flipping. Our training protocols followed previous works [21, 23, 66, 67]. Training is conducted for 100 epochs with a batch size of 256, using the SGD optimizer with a learning rate initially set to 0.1 and reduced by a factor of 10 at epochs 30, 60, and 90. A weight decay of 0.0001 and a momentum of 0.9 are applied to ensure stable convergence. This pretraining setup follows common practices from previous works to improve downstream performance on semantic segmentation tasks.

4.2.2 Training

The model is trained almost the same as previous works [15, 23, 33, 56, 59, 67, 70, 71] using the SGD optimizer with an initial learning rate of 0.008, momentum of 0.9, and weight decay of 0.0005. The learning rate follows a poly decay policy with a power of 0.9. Data augmentation includes random cropping to 1024×1024, horizontal flipping, and scaling within a range of 0.5 to 2.0. Training is conducted for 484 epochs (approximately 120K iterations) with a batch size of 10. For CamVid [3], the training process is similar to that of Cityscapes but runs for 200 epochs with a batch size of 24 and a learning rate of 0.003. Online Hard Example Mining (OHEM) [55] is also applied for more challenging sample selection, ensuring fair comparison with prior works.

doi:10.6342/NTU202500107

4.2.3 Measurement

We evaluate the inference speed on a platform equipped with a single NVIDIA RTX 3090 GPU, leveraging PyTorch 2.4 for the deep learning framework, CUDA 12.1 for GPU acceleration, and Ubuntu 20.04 as the operating system. To ensure consistent and comparable measurements of inference speed, we configure the batch size to 1, focusing on the performance of the model in processing individual data samples.

4.3 Comparison

Our proposed architecture outperforms existing methods like PIDNet [67] achieving state-of-the-art results.

4.3.1 Comparison without pretraining

Quantitative comparisons of model performance without pretraining are summerized in Table 4.1. Our proposed model surpasses state-of-the-art (SOTA) performance when both our model and SOTA models are trained without pretraining on ImageNet [52], underscoring its reduced reliance on large pretraining datasets. Notably, we outperform both PIDNet-M [67] and PIDNet-L [67] under these conditions. Despite PIDNet-L [67] requiring more time than our BEVANet, our model achieves superior performance, demonstrating the robustness and efficiency of our approach.

In practical scenarios where training efficiency is paramount, our model excels by training directly on the target dataset, achieving an impressive mIoU of nearly 79.3% with minimal preprocessing. This direct training approach significantly reduces the time

doi:10.6342/NTU202500107

and resources typically needed for pretraining and fine-tuning. The efficiency of our method accelerates the training process and eliminates the necessity for extensive pretraining datasets, making it a more practical and accessible solution for real-world applications with limited data availability and strict time constraints.

Model	FPS ↑	#Params (M) ↓	mIoU (%) ↑
PIDNet-S [67]	**93.2**	**7.6**	76.32
PIDNet-M [67]	39.8	34.4	78.22
PIDNet-L [67]	31.1	36.9	78.25
BEVANet (Ours)	32.9	58.62	**79.27**

Table 4.1: **Quantitative Comparisons of Model Performance without Pretraining.**

4.3.2 Overall Comparison

Model	Resolusion	GPU	FPS ↑	#Params (M) ↓	mIoU (%) ↑
DF2-Seg1 [36]	1536 × 768	GTX 1080Ti	67.2	-	75.9
DF2-Seg2 [36]	1536 × 768	GTX 1080Ti	56.3	-	76.9
BiSeNet(Res18) [70]	1536 × 768	GTX 1080Ti	65.5	49	74.8
BiSeNetV2-L [70]	1024 × 512	GTX 1080Ti	47.3	-	75.8
STDC1-Seg75 [15]	1536 × 768	RTX 3090	74.8	-	74.5
STDC2-Seg75 [15]	1536 × 768	RTX 3090	58.2	-	77.0
PP-LiteSeg-T2 [49]	1536 × 768	RTX 3090	96.0	-	76.0
PP-LiteSeg-B2 [49]	1536 × 768	RTX 3090	68.2	-	78.2
HyperSeg-M [44]	1024 × 512	RTX 3090	59.1	10.1	76.2
HyperSeg-S [44]	1536 × 768	RTX 3090	45.7	10.2	78.2
SwiftNetRN-18 [45]	2048 × 1024	GTX 1080Ti	39.9	11.8	75.5
CABiNet [31]	2048 × 1024	GTX 2080Ti	76.5	**2.64**	76.6
SFNet(DF2) [35]	2048 × 1024	RTX 3090	87.6	10.53	77.8
SFNet(ResNet-18) [35]	2048 × 1024	RTX 3090	30.4	12.87	78.9
DDRNet-23-S [23]	2048 × 1024	RTX 3090	**108.1**	5.7	77.8
DDRNet-23 [23]	2048 × 1024	RTX 3090	51.4	20.1	79.5
PIDNet-S-Simple [67]	2048 × 1024	RTX 3090	100.8	7.6	78.8
PIDNet-S [67]	2048 × 1024	RTX 3090	93.2	7.6	78.8
PIDNet-M [67]	2048 × 1024	RTX 3090	39.8	34.4	80.1
BEVANet (Ours)	2048 × 1024	RTX 3090	32.8	58.62	**81.0**

Table 4.2: **Overall Quantitative Comparisons on Cityscapes [11].**

Following pretraining on ImageNet [52] for a fair comparison, the results in Table 4.2 demonstrate that our BEVAN model consistently maintains a frame rate exceeding the real-time threshold of 30 FPS while delivering state-of-the-art performance on the

doi:10.6342/NTU202500107

Cityscapes dataset [11]. Our model outperforms all models of the same scale, achieving 81% mIoU. This underscores BEVAN's ability to balance high accuracy with real-time processing speed effectively. By optimizing advanced attention mechanisms and adaptive feature fusion, BEVAN sets a new benchmark by achieving the best semantic segmentation performance within the 30 FPS constraint, ensuring both precision and efficiency. Our approach proves that attaining top-tier performance is possible without sacrificing lots of processing speed.

Model	GPU	FPS ↑	mIoU (%) ↑
PP-LiteSeg-T [49]	GTX 1080Ti	154.8	75.0
BiSeNetV2 [70]	GTX 1080Ti	124.0	76.7
BiSeNetV2-L [70]	GTX 1080Ti	33.0	78.5
DDRNet-23-S [23]	RTX 3090	**182.4**	78.6
DDRNet-23 [23]	RTX 3090	116.8	80.6
PIDNet-S [67]	RTX 3090	153.7	80.1
PIDNet-S-Wider [67]	RTX 3090	85.6	82.0
BEVANet-S (Ours)	RTX 3090	79.4	**83.1**

Table 4.3: **Quantitative Comparisons on CamVid [3].**

Table 4.3 shows small-scale BEVAN also reaches SoTA on CamVid [3]. It reaches over 83% mIoU.

4.4 Ablation Study

4.4.1 Architecture Efficiency

As Table 4.4 demonstrates, our proposed architecture enhances speed with an acceptable reduction in performance. Designed for efficiency, our backbone achieves a

doi:10.6342/NTU202500107

significant boost in processing speed while maintaining a balanced trade-off in mIoU performance. The slight reduction in accuracy falls well within tolerable limits, making it suitable for real-time applications. It demonstrates the efficiency of processing through semantic concept and spatial detail branches.

Model	FPS ↑	#Params (M) ↓	mIoU (%) ↑
PIDNet [67]	42.64	29.22	**78.22**
BEVANet (Ours)	**44.25**	**28.31**	77.77

Table 4.4: **Quantitative Comparisons of Ablation Study on Architecture.**

4.4.2 Large Kernel Attention

By integrating the strengths of LSKA [32] and SLaK [40], we have developed the Sparse Decomposed Large Separable Kernel Attention (SDLSKA) block, an efficient and highly effective component for large kernel attention. In the Table 4.5, While LSKA [32] and SLaK [40] provide advancements in utilizing large receptive fields, our SDLSKA block surpasses both in accuracy and adaptability. LSKA [32] and SLaK [40] yield less than a 0.2% increase in mIoU, whereas our proposed method delivers an improvement of over 0.8%, achieving a performance of more than 78.5% mIoU. Proving SDLSKA captures and integrates the most valuable information.

Although there is a slight reduction in speed, the impact is minimal and remains suitable for real-time applications. More importantly, SDLSKA delivers a substantial performance boost of 0.82% mIoU, representing a remarkable leap forward in the field of real-time semantic segmentation. This improvement highlights the ability of SDLSKA to effectively capture and utilize multi-scale features and global context.

doi:10.6342/NTU202500107

Block	FPS ↑	#Params (M) ↓	mIoU (%) ↑
Convs	**44.25**	**28.31**	77.77
SLaK	37.76	45.25	77.84
LSKA	41.16	44.17	77.93
SDLSKA (Ours)	37.79	44.22	**78.60**

Table 4.5: **Quantitative Comparisons of Ablation Study on Large Kernel Attention.**

4.4.3 Selection Kernel

Our CKS module achieves a 0.26% mIoU improvement over LSKNet [37] with only a 0.5 FPS drop, demonstrating its efficient multi-scale kernel fusion. Unlike conventional methods restricted to square kernels, our approach seamlessly integrates one small square kernel and two strip dilation kernels, effectively balancing spatial and channel information. As shown in Table 4.6, our module excels in fusing three distinct feature types simultaneously, a task that requires a more advanced mechanism than traditional two-feature fusion. This adaptive fusion strategy ensures that neither spatial-wise nor channel-wise information is overlooked, capturing intricate dependencies across feature scales and shapes. By leveraging multi-scale kernel integration, our method enhances semantic segmentation performance while maintaining computational efficiency, making it a compelling solution for real-time applications.

Selection Kernel	FPS ↑	#Params (M) ↓	mIoU (%) ↑
Addition	**37.79**	**44.22**	78.60
LSKA[32]	37.46	48.18	78.72
CKS (Ours)	37.29	45.02	**78.86**

Table 4.6: **Quantitative Comparisons of Ablation Study on Selection Kernel.**

doi:10.6342/NTU202500107

4.4.4 Branch Fusion

Quantitative comparisons of ablation study on branch fusion are summarized in Table 4.7. Our thoughtful Boundary Guided Attention Fusion (BGAF) module outperforms BAG [67] by approximately 0.4% mIoU, underscoring the significance of its design choices. This improvement highlights the critical role of introducing a shortcut connection and adaptively processing low-level detailed spatial features and high-level semantic context features before their fusion, ensuring more precise feature representation and effective branch fusion.

Branch Fusion	FPS ↑	#Params (M) ↓	mIoU (%) ↑
Bag [67]	37.29	45.02	78.86
Light-Bag [67]	**38.48**	**44.56**	78.39
BGAF(Ours)	32.85	46.26	**79.27**

Table 4.7: **Quantitative Comparisons of Ablation Study on Branch Fusion.**

4.4.5 Multi-scale Fusion

As the comparisons in 4.8, our DLKPPM significantly enhances contextual fusion, delivering a much richer context compared to conventional models [23]. It achieves a 0.5% improvement in mean mIoU while sacrificing only 0.4 FPS, highlighting its exceptional efficiency. This performance boost is primarily due to the enlargement of the receptive field, which allows the model to refine features more precisely, capture semantic concepts more effectively, and reduce pooling information loss. By integrating larger receptive fields, the module excels in complex tasks, offering a robust solution for scenarios requiring real-time processing and high accuracy.

doi:10.6342/NTU202500107

PPM	FPS ↑	#Params (M) ↓	mIoU (%) ↑
DAPPM [23]	32.85	**32.44**	80.43
PAPPM [67]	**33.38**	**32.44**	79.97
DLKAPPM (Ours)	32.47	58.82	**80.96**

Table 4.8: **Quantitative Comparisons of Ablation Study on Pyramid Pooling Module.**

4.4.6 Overall without pretraining

Table 4.9 demonstrates the results of gradually replacing each block with our proposed block. Our model incorporates additional components sequentially, with each block contributing to improved performance while introducing a slight reduction in processing speed. This trade-off is justified by the substantial improvement in overall system efficiency. We believe the benefits gained from the increased performance outweigh the minor slowdown, making the deal well worth the investment. Under the 30 FPS constraint, our proposed modules boost the mIoU to nearly 79.3%, an increase of over 1.5%, representing a significant improvement. This underscores the critical impact and effectiveness of each block in our model.

Architecture	Block	Selection Kernel	Branch Fusion	FPS ↑	mIoU (%) ↑
PIDNet [67]	Convs	-	Bag [67]	42.64	78.22
BA (Ours)	Convs	-	Bag [67]	**44.25**	77.77
BA (Ours)	SLaK[40]	-	Bag [67]	37.76	77.84
BA (Ours)	LSKA[32]	-	Bag [67]	41.16	77.93
BA (Ours)	SDLSKA (Ours)	Addition	Bag [67]	37.79	78.60
BA (Ours)	SDLSKA (Ours)	LSKNet [37]	Bag [67]	37.46	78.72
BA (Ours)	SDLSKA (Ours)	CKS (Ours)	Bag [67]	37.29	78.86
BA (Ours)	SDLSKA (Ours)	CKS (Ours)	Light_Bag [67]	38.48	78.39
BA (Ours)	SDLSKA (Ours)	CKS (Ours)	BGAF (Ours)	32.85	**79.27**

Table 4.9: **The ablation study comparisons of our modules without pretraining.**

doi:10.6342/NTU202500107

4.5 Visualization

4.5.1 Small Object

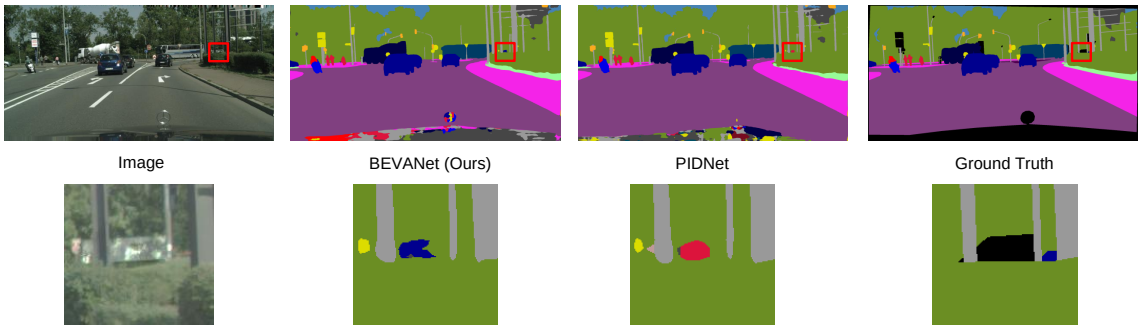

Figure 4.1: **Visualization Comparison for Small Objects.** Part1.

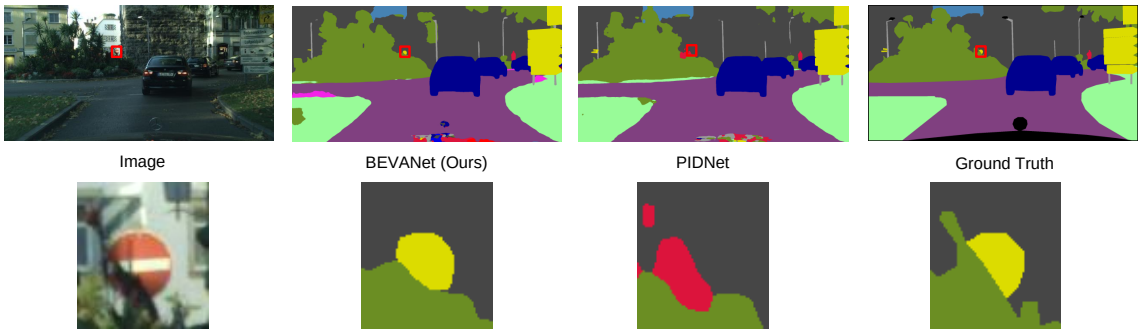

Figure 4.2: **Visualization Comparison for Small Objects.** Part2.

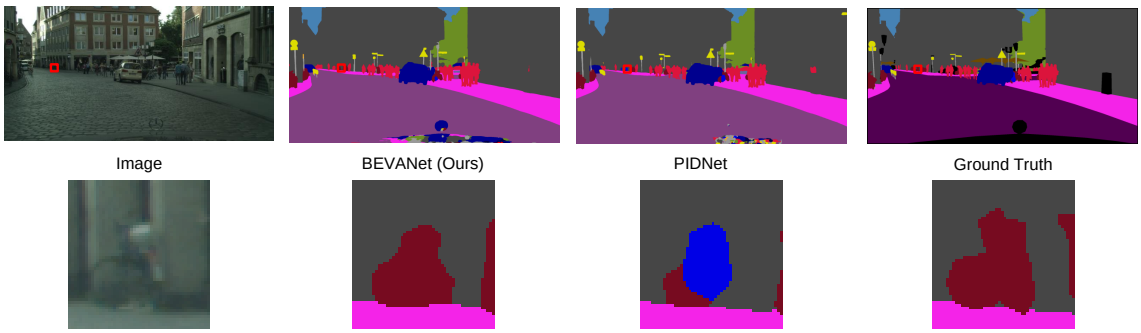

Figure 4.3: **Visualization Comparison for Small Objects.** Part3.

Our model demonstrates superior performance compared to PIDNet [67] in small object detection, which are inherently more difficult to identify. For instance, Figure 4.1 shows BEVAN accurately detects a small car, whereas PIDNet [67] mislabels it as a person. Similarly, Figure 4.2 tells our model identifies a small traffic sign with precision, while PIDNet [67] erroneously classifies it as a person. Figure 4.3 indicates our model

doi:10.6342/NTU202500107

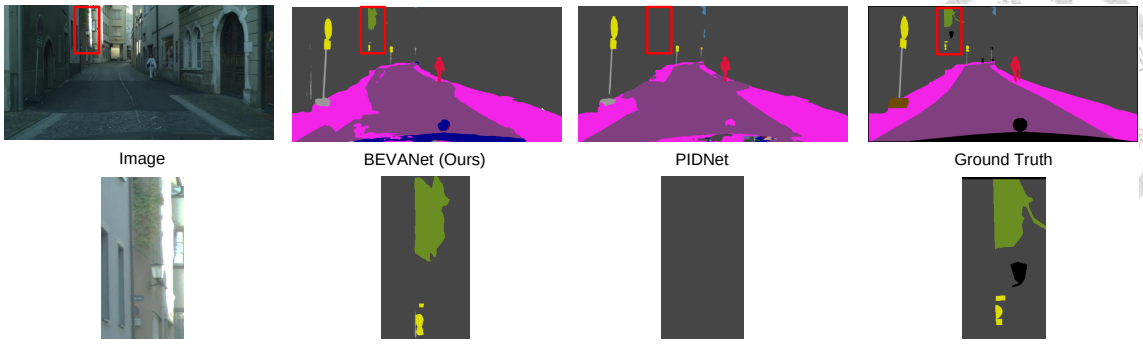

Figure 4.4: **Visualization Comparison for Small Objects.** Part4.

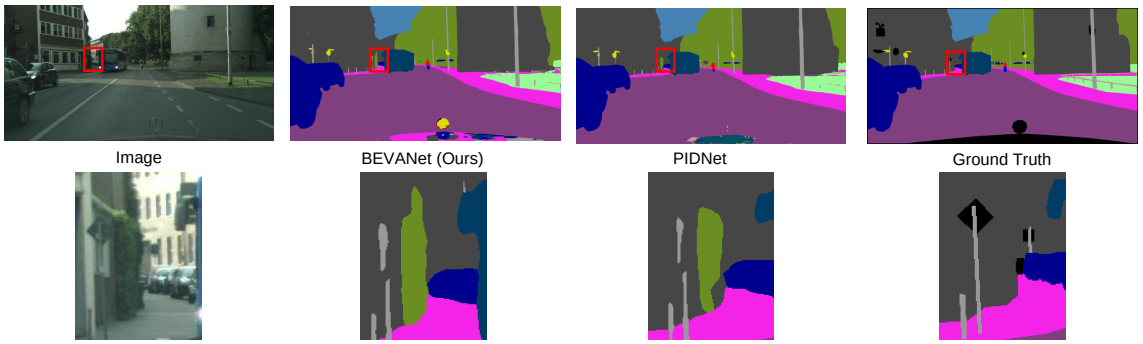

Figure 4.5: **Visualization Comparison for Small Objects.** Part5.

also correctly classifies a small bicycle, while PIDNet [67] misidentifies it as a car. In addition, Figure 4.4 implies our model successfully detects both small vegetation and the traffic sign, areas where PIDNet [67] fails to identify any objects at all. These instances clearly highlight that our model excels at detecting small objects, outperforming PIDNet [67] by a significant margin.

Moreover, Figure 4.5 demonstrates our model's ability to accurately identify small vegetation—despite the ground truth labeling it as a building—further underscores its capacity to capture the underlying semantic meaning of objects. This ability allows our model to make accurate predictions even when the ground truth fails to provide the correct label, demonstrating robust semantic understanding, accurate predictions, and a deeper comprehension of contextual information.

doi:10.6342/NTU202500107

4.5.2 Completeness

Our BEVAN leverages a large receptive field for thorough and accurate object detection, capturing entire objects with well-defined boundaries. Figure 4.6 shows that, when handling larger objects, our model excels at capturing them completely and with precise boundaries. This enables our model to outperform PIDNet [67], which struggles with object detection and boundary precision. For example, Figure 4.7 indicates our model consistently and reliably detects the presence of a sidewalk, accurately identifying it in various contexts. In contrast, PIDNet [67] rarely detects the sidewalk correctly, showcasing a notable gap in performance. This underscores our model's superior ability to capture spatial information and deliver more reliable object segmentation in complex environments.

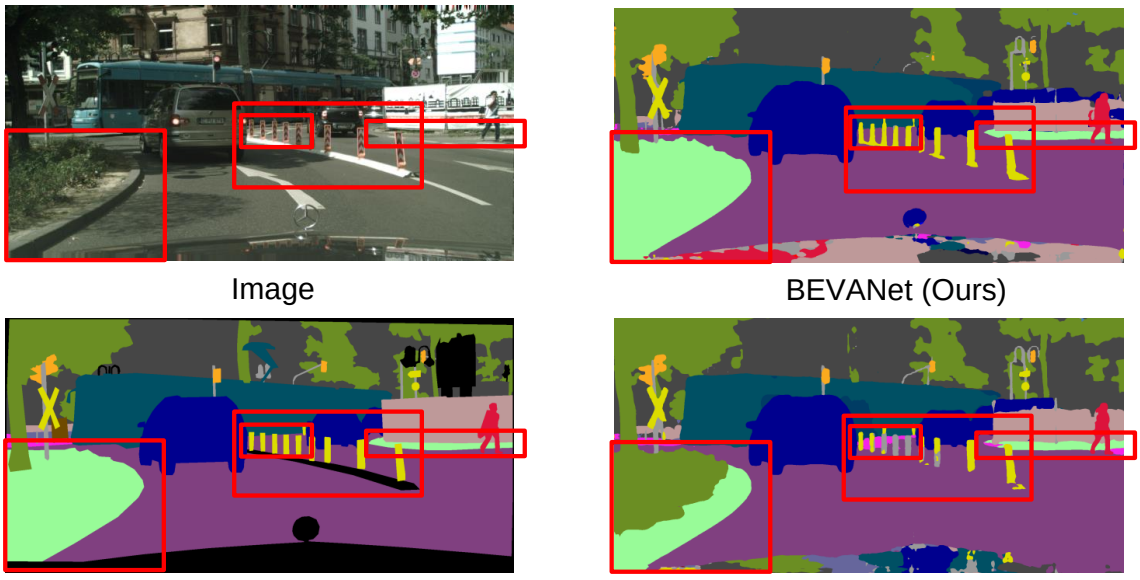

Figure 4.6: **Visualization Comparison for Completeness.** Part1.

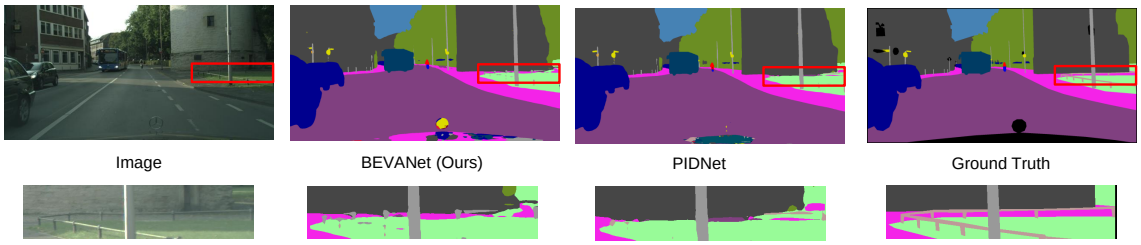

Figure 4.7: **Visualization Comparison for Completeness.** Part2.

doi:10.6342/NTU202500107

Chapter 5 Conclusion

Our BEVAN model achieves competitive performance compared to state-of-the-art methods while reaching real-time processing at 32 FPS. Its key features, including the Sparse Decomposed Large Separable Kernel Attentions (SDLSKA) block for effectively expanding the receptive fields to capture semantic context and the Comprehensive Kernel Selection (CKS) mechanism for dynamic receptive field adjustments by integrating features from both small and large kernels through channel and spatial attention, enable accurate small object detection and refined boundaries. The bilateral architecture communicates efficiently between feature levels, and the Boundary Guided Attention Fusion (BGAF) module further enhances feature fusion from different branches. Additionally, our Deep Large Kernel Pyramid Pooling Module (DLKPPM) enriches feature representations. Future work will focus on optimizing fusion strategies and reducing computational overhead to develop a lightweight large kernel attention model to further improve large kernel attention model efficiency.

doi:10.6342/NTU202500107

doi:10.6342/NTU202500107

References

[1] S. Asgari Taghanaki, K. Abhishek, J. P. Cohen, J. Cohen-Adad, and G. Hamarneh. Deep semantic segmentation of natural and medical images: a review. Artificial Intelligence Review, 54:137–178, 2021.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12):2481–2495, 2017.

[3] G. J. Brostow, J. Fauqueur, and R. Cipolla. Semantic object classes in video: A high-definition ground truth database. Pattern recognition letters, 30(2):88–97, 2009.

[4] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In Proceedings of the IEEE/CVF international conference on computer vision workshops, pages 0–0, 2019.

[5] L.-C. Chen. Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint arXiv:1412.7062, 2014.

[6] L.-C. Chen. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587, 2017.

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab:

doi:10.6342/NTU202500107

Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence, 40(4):834–848, 2017.

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

[9] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu. Dynamic convolution: Attention over convolution kernels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 11030–11039, 2020.

[10] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1251–1258, 2017.

[11] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3213–3223, 2016.

[12] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. In Proceedings of the IEEE international conference on computer vision, pages 764–773, 2017.

[13] X. Ding, X. Zhang, J. Han, and G. Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 11963–11975, 2022.

doi:10.6342/NTU202500107

[14] A. Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.

[15] M. Fan, S. Lai, J. Huang, X. Wei, Z. Chai, J. Luo, and X. Wei. Rethinking bisenet for real-time semantic segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 9716–9725, 2021.

[16] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, and K. Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. IEEE Transactions on Intelligent Transportation Systems, 22(3):1341–1360, 2020.

[17] M. Gamal, M. Siam, and M. Abdel-Razek. Shuffleseg: Real-time semantic segmentation network. arXiv preprint arXiv:1803.03816, 2018.

[18] M.-H. Guo, C.-Z. Lu, Q. Hou, Z. Liu, M.-M. Cheng, and S.-M. Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. Advances in Neural Information Processing Systems, 35:1140–1156, 2022.

[19] M.-H. Guo, C.-Z. Lu, Z.-N. Liu, M.-M. Cheng, and S.-M. Hu. Visual attention network. Computational Visual Media, 9(4):733–752, 2023.

[20] D. Haase and M. Amthor. Rethinking depthwise separable convolutions: How intra-kernel correlations lead to improved mobilenets. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 14600–14609, 2020.

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

doi:10.6342/NTU202500107

[22] D. Hendrycks and K. Gimpel. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415, 2016.

[23] Y. Hong, H. Pan, W. Sun, and Y. Jia. Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes. arXiv preprint arXiv:2101.06085, 2021.

[24] Q. Hou, C.-Z. Lu, M.-M. Cheng, and J. Feng. Conv2former: A simple transformer-style convnet for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024.

[25] Y. Hou, Z. Ma, C. Liu, and C. C. Loy. Learning lightweight lane detection cnns by self attention distillation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 1013–1021, 2019.

[26] A. G. Howard. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.

[27] H. Hu, Z. Zhang, Z. Xie, and S. Lin. Local relation networks for image recognition. In Proceedings of the IEEE/CVF international conference on computer vision, pages 3464–3473, 2019.

[28] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi. Gather-excite: Exploiting feature context in convolutional neural networks. Advances in neural information processing systems, 31, 2018.

[29] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7132–7141, 2018.

doi:10.6342/NTU202500107

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 2012.

[31] S. Kumaar, Y. Lyu, F. Nex, and M. Y. Yang. Cabinet: Efficient context aggregation network for low-latency semantic segmentation. In 2021 IEEE International Conference on Robotics and Automation (ICRA), pages 13517–13524. IEEE, 2021.

[32] K. W. Lau, L.-M. Po, and Y. A. U. Rehman. Large separable kernel attention: Rethinking the large kernel attention design in cnn. Expert Systems with Applications, 236:121352, 2024.

[33] H. Li, P. Xiong, H. Fan, and J. Sun. Dfanet: Deep feature aggregation for real-time semantic segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 9522–9531, 2019.

[34] X. Li, W. Wang, X. Hu, and J. Yang. Selective kernel networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 510–519, 2019.

[35] X. Li, A. You, Z. Zhu, H. Zhao, M. Yang, K. Yang, S. Tan, and Y. Tong. Semantic flow for fast and accurate scene parsing. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16, pages 775–793. Springer, 2020.

[36] X. Li, Y. Zhou, Z. Pan, and J. Feng. Partial order pruning: for best speed/accuracy trade-off in neural architecture search. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 9145–9153, 2019.

doi:10.6342/NTU202500107

[37] Y. Li, Q. Hou, Z. Zheng, M.-M. Cheng, J. Yang, and X. Li. Large selective kernel network for remote sensing object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 16794–16805, 2023.

[38] A. Lin, B. Chen, J. Xu, Z. Zhang, G. Lu, and D. Zhang. Ds-transunet: Dual swin transformer u-net for medical image segmentation. IEEE Transactions on Instrumentation and Measurement, 71:1–15, 2022.

[39] J.-J. Liu, Q. Hou, M.-M. Cheng, C. Wang, and J. Feng. Improving convolutional networks with self-calibrated convolutions. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10096–10105, 2020.

[40] S. Liu, T. Chen, X. Chen, X. Chen, Q. Xiao, B. Wu, T. Kärkkäinen, M. Pechenizkiy, D. Mocanu, and Z. Wang. More convnets in the 2020s: Scaling up kernels beyond 51x51 using sparsity. arXiv preprint arXiv:2207.03620, 2022.

[41] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF international conference on computer vision, pages 10012–10022, 2021.

[42] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie. A convnet for the 2020s. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 11976–11986, 2022.

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

doi:10.6342/NTU202500107

[44] Y. Nirkin, L. Wolf, and T. Hassner. Hyperseg: Patch-wise hypernetwork for real-time semantic segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 4061–4070, 2021.

[45] M. Orsic, I. Kreso, P. Bevandic, and S. Segvic. In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 12607–12616, 2019.

[46] K. O'Shea. An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458, 2015.

[47] J. Park. Bam: Bottleneck attention module. arXiv preprint arXiv:1807.06514, 2018.

[48] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. arXiv preprint arXiv:1606.02147, 2016.

[49] J. Peng, Y. Liu, S. Tang, Y. Hao, L. Chu, G. Chen, Z. Wu, Z. Chen, Z. Yu, Y. Du, et al. Pp-liteseg: A superior real-time semantic segmentation model. arXiv preprint arXiv:2204.02681, 2022.

[50] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. IEEE Transactions on Intelligent Transportation Systems, 19(1):263–272, 2017.

[51] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18, pages 234–241. Springer, 2015.

doi:10.6342/NTU202500107

[52] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. International journal of computer vision, 115:211–252, 2015.

[53] M. Saha and C. Chakraborty. Her2net: A deep framework for semantic segmentation and classification of cell membranes and nuclei in breast cancer evaluation. IEEE Transactions on Image Processing, 27(5):2189–2200, 2018.

[54] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4510–4520, 2018.

[55] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 761–769, 2016.

[56] T.-H. Tsai and Y.-W. Tseng. Bisenet v3: Bilateral segmentation network with coordinate attention for real-time semantic segmentation. Neurocomputing, 532:33–42, 2023.

[57] Q. Wan, Z. Huang, J. Lu, Y. Gang, and L. Zhang. Seaformer: Squeeze-enhanced axial transformer for mobile semantic segmentation. In The eleventh international conference on learning representations, 2023.

[58] J. Wang, C. Gou, Q. Wu, H. Feng, J. Han, E. Ding, and J. Wang. Rtformer: Efficient design for real-time semantic segmentation with transformer. Advances in Neural Information Processing Systems, 35:7423–7436, 2022.

[59] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, et al. Deep high-resolution representation learning for visual recognition.

doi:10.6342/NTU202500107

IEEE transactions on pattern analysis and machine intelligence, 43(10):3349–3364, 2020.

[60] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF international conference on computer vision, pages 568–578, 2021.

[61] Z. Wang, X. Lin, N. Wu, L. Yu, K.-T. Cheng, and Z. Yan. Dtmformer: Dynamic token merging for boosting transformer-based medical image segmentation. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pages 5814–5822, 2024.

[62] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, and S. Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 16133–16142, 2023.

[63] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon. Cbam: Convolutional block attention module. In Proceedings of the European conference on computer vision (ECCV), pages 3–19, 2018.

[64] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang. Cvt: Introducing convolutions to vision transformers. In Proceedings of the IEEE/CVF international conference on computer vision, pages 22–31, 2021.

[65] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in neural information processing systems, 34:12077–12090, 2021.

doi:10.6342/NTU202500107

[66] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1492–1500, 2017.

[67] J. Xu, Z. Xiong, and S. P. Bhattacharyya. Pidnet: A real-time semantic segmentation network inspired by pid controllers. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 19529–19539, 2023.

[68] B. Yang, G. Bender, Q. V. Le, and J. Ngiam. Condconv: Conditionally parameterized convolutions for efficient inference. Advances in neural information processing systems, 32, 2019.

[69] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang. Denseaspp for semantic segmentation in street scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3684–3692, 2018.

[70] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. International journal of computer vision, 129:3051–3068, 2021.

[71] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In Proceedings of the European conference on computer vision (ECCV), pages 325–341, 2018.

[72] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha, et al. Resnest: Split-attention networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2736–2746, 2022.

doi:10.6342/NTU202500107

[73] W. Zhang, Z. Huang, G. Luo, T. Chen, X. Wang, W. Liu, G. Yu, and C. Shen. Topformer: Token pyramid transformer for mobile semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12083–12093, 2022.

[74] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 6848–6856, 2018.

[75] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. Icnet for real-time semantic segmentation on high-resolution images. In Proceedings of the European conference on computer vision (ECCV), pages 405–420, 2018.

[76] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2881–2890, 2017.

[77] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 6881–6890, 2021.

[78] X. Zhu, H. Hu, S. Lin, and J. Dai. Deformable convnets v2: More deformable, better results. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 9308–9316, 2019.

doi:10.6342/NTU202500107