Learning Discriminative Neural Representations for Visual Recognition

Sudong Cai

Acknowledgements

My life in Kyoto is a wonderful journey. I would like to sincerely thank the following people who have helped and supported me in my research, study, and daily life.

I would first like to express my heartfelt thanks to Professor Ko Nishino sensei, my supervisor. Nishino sensei offered me the precious opportunity to work and study as a Ph.D. student in Nishino lab at Kyoto. Nishino sensei's advice and teaching are of particular significance to my growing up as a researcher with the necessary research skills.

I am grateful to have the rest of my doctoral committees: Professor Tatsuya Akutsu sensei and Professor Hisashi Kashima sensei. I would like to sincerely thank their kindness in taking out their valuable time to participate in my Ph.D. dissertation defense.

I would like to express my sincere gratitude to Professor Shohei Nobuhara sensei. Nobuhara sensei is always kind and patient. His valuable advice and kind help supported me a lot in my study and life in Kyoto.

I would like to sincerely thank Asako Yoshimura san. Without her kind help, I might have encountered various troubles in my study and life overseas.

I am really grateful to have the opportunity to know my colleagues in Nishino lab. I would express my particular thanks to Marc A. Kastner sensei, Kohei Yamashita kun, Shu Nakamura kun, Meng-Yu Jennifer Kuo kun, Ryosuke Wakaki kun, Sangeun Lee san, Sicheng He kun, Zhe Chen san, Yuzheng Xu kun, Yang Wu sensei, Mai Nishimura san, and Nicole Xinran Han kun. They helped and supported me so much in my life, research, and study. I am also appreciate the chance to know Yuta Yoshitake kun, Yiming Shi kun, Shi Chen kun, Ryo Kawahara san, Soma Nonaka kun, Yupeng Liang kun, Keisuke Shibata kun, and other junior lab members.

I want to thank the Student Affairs Office and the International Student Support Section. They always be so patient to help me. I would particularly thank MEXT for offering me the scholarship that supported me enormously during these years in Kyoto.

I want to express my special thanks to my family, especially my wife Chunting Liu. She has always been there for me, no matter what difficulties I have met. Her support, encouragement, and promise are priceless treasures in my life. If she weren't here, my life would be totally different. I also want to thank my parents sincerely. They helped me a lot to overcome the difficulties in my life.

All in all, I am sincerely thankful for these three years in Kyoto, it has become an invaluable story in my life and left me with so many lovely memories.

Abstract

Neural representations generated by neural networks are the foundation for the unprecedented success of deep-learning-based visual recognition. Self-attention and self-gating are relevant mechanisms inspired by neuronal behaviors. They contribute to learning effective neural representations from images. **S**elf-**A**ttention (**SA**) models the fine-grained weights of features based on the relative feature similarities and generates the context-aware attended features by integrating the re-weighted features dynamically. Self-gating instead calculates feature weights with the local/non-local cues with a non-linear curve that introduces oriented inductive biases. It enables static yet efficient feature re-calibration.

We investigate learning effective neural representations for visual recognition tasks by leveraging self-attention and self-gating in three new approaches. We first address RGB **R**oad scene **M**aterial **S**egmentation (**RMS**), an unstudied problem of particular importance for scene understanding. We identify the encoding and fusion of multi-scale texture cues and image context as the key to accurate RGB RMS. We propose **RMSNet** framework built on **SAMixer**, a novel context-aware multi-scale decoder for effective and efficient feature fusion. SAMixer extends the original spatial SA to a feature aggregator that mixes multi-level features at each aligned pixel location by introducing a highly efficient query-key similarity measure tailored to many-to-one feature fusion, namely, *Balanced Q-K Sim*. We validate SAMixer with extensive experiments on the RGB RMS benchmark.

Inspired by the observation that the same image can have different annotations of semantic objects and materials, simultaneously, we investigate improved self-gated neural feature activation for general visual recognition in images. We begin by rethinking neural activation from **M**ulti-**C**riteria **D**ecision-**M**aking (**MCDM**), where we treat activation models as selective re-calibrators that suppress/emphasize features according to their importance scores measured by feature-filter similarities. This helps us identify the unexcavated yet critical problem of *mismatched feature scoring* led by the differentiated feature/filter norms and inspires our two novel activation prototypes, namely, **I**nstantaneous **I**mportance **E**stimation **U**nit (**IIEU**) and **AdaShift**. We propose IIEU as the first solution to the problem *mismatched feature scoring*, which re-calibrates features with the **II**-score estimated with the adaptive norm-decoupled feature-filter similarities, capable of modeling the inter-channel feature-filter relationships at a low cost. We then introduce AdaShift as an efficiency-boosted solution to address norm-induced biases. AdaShift casts dynamic translations on the inputs of the re-weighting function by an **Ada**ptive **Shift** factor that exploits feature-filter contextual cues of different ranges in a simple yet effective manner. We obtain the intuitions of

AdaShift by rethinking the feature-filter relationships from a common Softmax-based classification. Our practical activation models built on IIEU and AdaShift prototypes, respectively, are validated through extensive experimental evaluations and quantitative analysis, where they demonstrate significant improvements to the popular/SOTA activation models on various vision benchmarks.

Keywords: Neural Networks, Visual Recognition, RGB Road Scene Material Segmentation, Neural Feature Activation, Self-Attention, Self-Gating, MCDM.

List of publications

Publications included in this dissertation

This dissertation consists of the following four publications.

Referred International Conference Proceedings

- (Chapter 3): Sudong Cai, Ryosuke Wakaki, Shohei Nobuhara, Ko Nishino, "RGB Road Scene Material Segmentation", In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, Springer Nature, pp. 3051–3067, 2022. [1]
  ©2022, ACCV, Springer Nature. https://link.springer.com/chapter/10.1007/978-3-031-26284-5_16

- (Chapter 4): Sudong Cai, "IIEU: Rethinking Neural Feature Activation from Decision-Making", In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5796–5806, 2023. [2]
  ©2023, ICCV, IEEE. https://doi.org/10.1109/ICCV51070.2023.00533

- (Chapter 5): Sudong Cai, "AdaShift: Learning Discriminative Self-Gated Neural Feature Activation With an Adaptive Shift Factor", *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, under review

Referred Journal Articles

- (Chapter 3): Sudong Cai, Ryosuke Wakaki, Shohei Nobuhara, Ko Nishino, "RGB Road Scene Material Segmentation", *Image and Vision Computing (IMAVIS)*, Elsevier, under review (major revision)

[1] ©2022 Springer Nature [1]. Reproduced with permission from Springer Nature. In reference to the corresponding Springer Nature Copyright Permission Guidelines (https://www.springernature.com/gp/partners/rights-permissions-third-party-distribution), Springer Nature Book and Journal Authors have the right to reuse the Version of Record, in whole or in part, in their own thesis. Additionally, authors may reproduce and make available their thesis, including Springer Nature content, as required by their awarding academic institution.

[2] ©2023 IEEE [2]. Reprinted with permission. In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of Kyoto University's products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to https://www.ieee.org/publications/rights/copyright-policy.html to learn how to obtain a License from RightsLink.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

1.1.1 Motivation and Goal

Neural representations extracted by neural networks significantly facilitate modern visual pattern recognition in images [30, 31, 32, 33]. Recently, the relevant biologically inspired mechanisms, *i.e*. self-attention [34, 35] and self-gating [36] are playing an increasingly important role in deep-learning-based vision applications [19, 26, 25, 2, 37, 18, 1]. They contribute to the learning of effective neural representations by enabling the *adaptive information selections* through feature re-weighting. In this dissertation, we extensively investigate two critical challenges in deep-learning-based visual recognition – (1) an unexplored special task of significant meaning for general scene understanding, namely, *RGB **R**oad scene **M**aterial **S**egmentation (**RMS**)* [1]; and (2) the general problem that touches the foundation of vision neural networks and neural representations, *i.e*., *neural feature activation with image inputs*, where we present three novel methods by leveraging self-attention and self-gating mechanisms, lay in the very original intention of neural attention and gating, *i.e*. discriminative yet low-cost.

1.1.2 Adaptive Neural Feature Selection: From A Specific to THE General

RGB road scene material segmentation [1], an unexplored problem of particular meaning for general scene understanding, is not yet another semantic segmentation task with just a

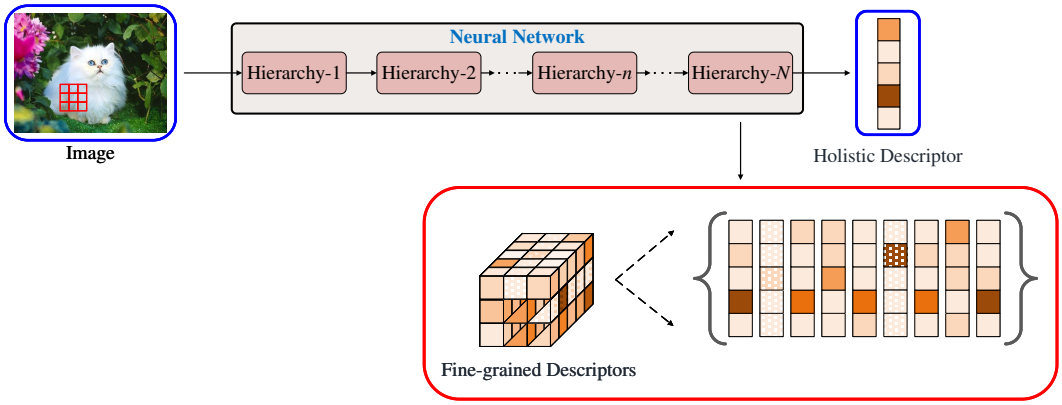


Figure 1.1: An illustration – although image-based visual recognition problems can be diverse, the basic goal of deep-learning-based approaches for visual recognition can be unified, *i.e.*, to learn the effective holistic descriptor that represents the objective image or the fine-grained descriptors that represent the detailed patterns.

different set of labels. This lies in the fact that materials, the raw ingredients of things, exhibit distinct visual properties to their composed objects, *e.g.*, it would be difficult to obtain effective shape cues of material categories as they often show fragmented spatial distributions in road scenes. In contrast, for a real-world object in a road scene, its signature shape can be a primary cue for recognition. Through careful analysis of the images with material annotations, we identify the effective fusion of multi-scale texture and context cues of materials, which enables the cross-enhancement of the two kinds of important information, as the key to producing discriminative neural representations of material appearances in road scenes. This inspires our **SAMixer** model that realizes discriminative context-aware multi-scale multi-level feature fusion through a novel efficient **M**ulti-head **S**elf-**A**ttention (**MSA**) mechanism. SAMixer significantly improves the current prevalent dense decoding heads, proposed for object segmentation, which also leverage multi-scale cues in feature fusion by the improved MSA-based selective feature fusion process.

Furthermore, despite the critical differences in properties of the material categories and the object categories, we aim to find their important commonality for deep-learning-based visual recognition. With this intention, our observation that *the same road scene image can naturally have different annotations of object categories and material categories, simultaneously*, triggers our new intuitions of modeling adaptive information selection on neural features for general visual recognition. A corresponding example is shown in Figure 1.2. Specifically, we suppose in supervised learning, a neural network encodes the discriminative representations through *learning to select the targeted information corresponding to the designated annotations* and identify that *nonlinear neural activation process, which helps fit the underlying mappings of objectives, is a key to such oriented information selection in each neuron*.

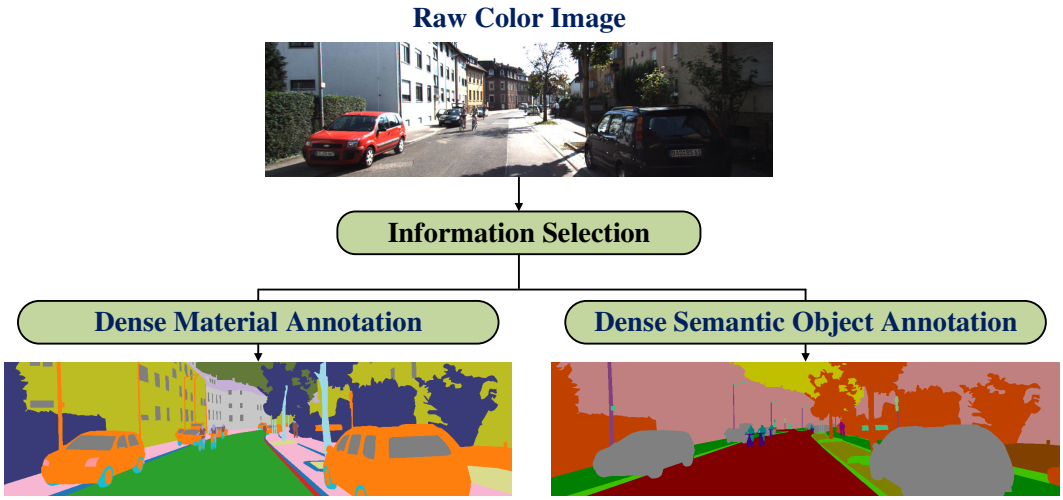


Figure 1.2: An illustration – the same road scene color image can naturally have different sets of annotation of material categories (left) and semantic object categories (right), simultaneously.

To embody this basic intuition, we propose to seek improved neural activation models by interpreting neural feature activation from a new generalized perspective of **M**ulti-**C**riteria **D**ecision-**M**aking (**MCDM**). In contrast to the prevalent activation functions, *e.g.*, ReLU [5] and Softplus [38] that originate from the specific behaviors of neuronal response [39, 40], as an MCDM problem, we instead treat activation functions as selective feature recalibrators that suppress/emphasize features according to their importance scores measured by the feature-filter similarities. This new interpretation enables us to bridge the meaning of "selectivity" and "nonlinearity" of neural activation through a simple proposition and identify the critical yet unexcavated problem, *mismatched feature scoring*, occurred in a typical neural activation process, led by the differentiated feature and filter (2-)norms, but invisible to past biological intuitions of neural activation. Based on the simple MCDM hypothesis, we introduce two novel self-gated neural activation prototypes by addressing the unsolved *mismatched feature scoring* problem with the corresponding new intuitions and assumptions, which we refer to as **IIEU** and **AdaShift**, respectively.

In Chapter 2, we briefly introduce the preliminary knowledge and concepts of our works in this dissertation, which includes (a) the basic self-attention and self-gating mechanisms for image data; (b) the basic multi-criteria decision-making methods that inspire our generalized MCDM hypothesis for neural activation.

1.2 RGB RMS With Selective Feature Fusion

We begin by addressing RGB road scene material segmentation, *i.e.*, per-pixel recognition of materials in real-world driving views with RGB images, an unstudied problem of particular importance for general scene understanding. Recognition of materials is critical for

real-world computer vision applications, since materials, what objects are made of, inform the informative physical attributes of semantic objects and regions in a scene that are unreachable by attending only to object categories. The significance of dense per-pixel RGB material recognition can be even higher for road scenes, especially for the safety and effective driving planning of autonomous vehicles that navigate in diverse traffic environments. The accurate dense segmentation of materials in road scenes, however, can be particularly challenging, which lies in the fact that the same road scene object is possibly made of very different materials, *e.g.*, roads can be built of "asphalt," "concrete," or even "brick," but they are indiscernible from shapes. This challenge can be exacerbated further by the fact that the surface of an individual object can be composed of various material ingredients, *e.g.*, a car composed of "metal," "glass," "rubber, vinyl," and "plastic." In contrast, characteristic shapes often serve as a type of primary cues for discerning semantic objects.

By carefully analyzing images from the new benchmark dataset focusing on RGB RMS, *i.e.* KITTI-Materials [1], we identify the effective extraction and fusion of local texture cues and image context of materials as the key to generating accurate representations of material appearance in road scenes. Specifically, the signature textures of materials inform discriminative cues for their identification. They, however, can change drastically with occlusion and scale, *i.e.* distance from viewpoint, which impedes the learning of robust neural representations. For object segmentation, the incorporation of structural dependencies helps to refine the texture cues by arriving at representations relatively robust to the scale and occlusion variations. However, directly generalizing this knowledge to materials can encounter significant challenges due to the weak and sensitive shape cues of typical road scene material regions. For this, we suppose that the effective use of the scale-aware fine-grained context of materials, which contributes to emphasizing the levels and scales of features that attend to the corresponding material textures, is even vital to learning discriminative joint neural representations for RGB RMS.

This intuition originates our **RMSNet** framework for accurate RGB RMS, constructed on the novel self-attention-based **SAMixer** model that enables effective context-aware multi-scale feature fusion with high efficiency. As the key idea, SAMixer generalizes the original MSA (with quadratic complexity) that integrates feature vectors on a spatial lattice to a feature aggregator that mixes multi-level features at each aligned pixel location by introducing a new balanced query-key similarity measure that realizes the first highly efficient many-to-one feature fusion through the pure Q-K-V attention process (to the best of our knowledge) with a single time of vector-matrix multiplication (*i.e.*, with linear complexity).

We validate the effectiveness of RMSNet and SAMixer with extensive experiments on the RGB RMS benchmark KITTI-Materials. The auxiliary evaluations on the relevant datasets MCubeS and images from Cityscapes demonstrate the generalizability of our RMSNet and SAMixer for RGB RMS on the realistic driving view images.

The details of this work, including the background, related works, method, and experimental results, are introduced in **Chapter 3**.

1.3 Modeling Discriminative Neural Activation With MCDM

Nonlinear activation functions are the foundation for the unprecedented success of neural networks in pattern recognition applications [30, 31, 32, 33]. The choice of the activation function is a decisive yet non-trivial factor in the performance of a neural network. Basic activation functions such as ReLU [5] and Softplus [38] are encouraged by the specific neuronal responses for stimulations [39, 40]. Based on the basic activation paradigms, past works have investigated to improve activation functions by leveraging channel/spatial feature context (*e.g.*, FReLU [41], Dynamic-ReLU [42] and ACONs [7]), statistical strategies (*e.g.*, GELU [12], Pserf [15], and Smooth-Maximum-Units [8]), and task-specific periodic functions [43, 44]. Existing methods of neural activation, however, still leave critical problems in the optimal decision on/design of practical activation functions. As a major reason, although several past efforts [45, 46, 47] suggested learning adaptive activation models with dynamic approximators, it still lacks tailored interpretations to help specify the properties of effective activation models for visual recognition. Such specific properties, however, are difficult to be identified from pure biological intuitions that underlie current prevalent activation functions.

To explore new improvements in neural feature activation, in contrast to the prevalent activation functions that impose nonlinear inductive biases by simulating the biological neuronal responses of primates, we instead regard neural feature activation as a generalized process of **M**ulti-**C**riteria **D**ecision-**M**aking (**MCDM**), where *an activation function is treated as a selective feature re-calibrator that suppresses or emphasizes features according to their importance scores measured by the feature-filter similarities*. This understanding helps us to bridge the meaning of "selectivity" and "nonlinearity" of neural activation through a simple proposition, identify the critical yet unexplored problem of *mismatched feature scoring* in a typical neural activation process led by the differentiated feature and filter norms, and introduce two novel self-gated neural activation prototypes by addressing the unsolved problem, namely, **I**nstantaneous **I**mportance **E**stimation **U**nit (**IIEU**) and **AdaShift**.

We propose IIEU as the first solution to the unstudied problem *mismatched feature scoring*, which we build from scratch by proposing a set of new assumed properties of an effective neural activation based on the new intuitions from our MCDM hypothesis. IIEU re-calibrates features with the **I**nstantaneous **I**mportance (**II**) score, which we refer to as, estimated with the adaptive norm-decoupled feature-filter similarities, capable of modeling the informative cross-channel feature-filter relationships at a low computational cost. Based on the IIEU prototype, we present IIEU-**B** (*i.e.*, -**B**asic) as the initial IIEU derivative for practical application and IIEU-**DC** (-**D**ynamic -**C**oupler) as a tailored enhancement to IIEU-B. The methodological details of IIEU-B and IIEU-DC are introduced in Section 4.3.2.

Furthermore, with the suggested MCDM hypothesis, we obtain new intuitions of self-gated neural activation by rethinking the feature-filter relationships from a common Softmax-based classification and by generalizing the new observations to a common learning layer

that encodes neural features with updatable filters, where we argue that *feature and filter norms can represent informative cues for neural feature activation* and the brute-force-style decoupling of feature/filter norms in a feature weight calculation possibly neutralizes the discriminativeness of the weight. These encourage our AdaShift as an efficiency-boosted solution to the *mismatched feature scoring* problem. AdaShift enhances a self-gated neural activation by incorporating an **Ada**ptive **Shift** factor into the re-weighting process of activation. It introduces dynamic translations to the inputs of the re-weighting function by an **Ada**ptive **Shift** factor that exploits the feature-filter context of different ranges in a simple yet effective manner. This enables AdaShift to cast flexible non-monotonic feature re-weighting by adapting to the current learning states. Based on the AdaShift prototype, we propose AdaShift-**B** as the basic practical AdaShift and AdaShift-**MA** (-**M**inimalist **A**ttention) as an enhanced practical AdaShift by introducing a **M**inimalist-style self-**A**ttention operation. The methodological details of AdaShift-B and AdaShift-MA are introduced in Section 5.4.2.

Our practical activation functions built on the novel IIEU and AdaShift prototypes, respectively, are validated through extensive experimental evaluations and quantitative analysis, where they demonstrate significant improvements to the popular/**S**tate-**O**f-**T**he-**A**rt (**SOTA**) activation functions on various vision benchmark datasets of different visual recognition tasks.

The details of these two works, including the backgrounds, related works, methods, and experimental results, are introduced in **Chapter 4** and **Chapter5**, respectively.

1.4 Contribution

This dissertation introduces our three pieces of research on deep-learning-based visual recognition, where the first work investigates the specific task, *RGB road scene material segmentation*, and the next two works explore the general problem, *neural feature activation with image inputs*. Below we summarize the main contributions of these three pieces of work, respectively.

The main contributions of our first work (titled "RGB Road Scene Material Segmentation") are threefold:

1. we investigate the unexplored yet significant visual recognition problem for general scene understanding, RGB road scene material segmentation, based on deep learning techniques;
2. we propose RMSNet, a new baseline deep learning framework for improving RGB RMS, built on the novel context-aware multi-level multi-scale fusion model, SAMixer, for enhancing neural features, especially for Transformer-induced features, which to the best of our knowledge, realizes the first highly efficient many-to-one feature fusion through the pure Q-K-V attention process for dense per-pixel recognition with linear complexity;

3. and we validate our RMSNet (SAMixer) through extensive quantitative analysis and ablation studies on the RGB RMS benchmark, *i.e.*, KITTI-Materials. Experimental comparisons with various relevant RGB material segmentation and road scene semantic segmentation methods clearly demonstrate the effectiveness. Moreover, the auxiliary evaluations on the relevant datasets MCubeS and images from Cityscapes verify the generalizability of our RMSNet (SAMixer) for RGB RMS on the realistic driving view images.

The main contributions of our second work (titled "IIEU: Rethinking Neural Feature Activation from Decision-Making") are threefold:

1. we suggest the MCDM hypothesis for neural feature activation, where we identify the unstudied yet critical problem of *mismatched feature scoring* in a typical neural activation process and introduce a set of new intuitions to help interpret the working mechanism of activation functions from a new generalized perspective of MCDM;
2. we introduce the novel activation prototype, IIEU, built from scratch on the suggested MCDM hypothesis, as the initial solution to the *mismatched feature scoring* problem;
3. we present the practical activation models, *i.e*. IIEU-B and IIEU-DC, based on the IIEU prototype and extensively validate (a) the effectiveness and versatility of IIEUs with various vision benchmark datasets, where IIEUs significantly improve the popular/SOTA activation functions; (b) our intuitions/hypothesis with targeted ablation studies.

The main contributions of our third work (titled "AdaShift: Learning Discriminative Self-Gated Neural Feature Activation With an Adaptive Shift Factor") are threefold:

1. we introduce the novel self-gated neural activation prototype, AdaShift, as an efficiency-boosted solution to the *mismatched feature scoring* problem.
2. based on the AdaShift prototype, we present the efficient practical activation functions, AdaShifts, which improve the prevalent self-gated activation functions significantly and also match/outperform the current SOTA, IIEU(s), with higher efficiency;
3. we extensively validate (a) the effectiveness and versatility of our practical AdaShifts with various vision benchmark datasets; (b) the extensibility and generalizability of our AdaShift prototype with targeted ablation studies and quantitative analysis.

Chapter 2

Preliminaries

In this Chapter, we briefly introduce the preliminary knowledge and concepts of our works in this dissertation, *i.e.*, (a) the basic self-attention and self-gating mechanisms for image data and (b) the basic multi-criteria decision-making methods that inspire our simple generalized **M**ulti-**C**riteria **D**ecision-**M**aking (**MCDM**) hypothesis for neural activation.

2.1 Basic Self-Attention and Self-Gating Mechanisms for Visual Recognition

The neuronal visual systems of primates share an extraordinary ability to interpret complex objects and scenes in real-time, despite the resource constraints of biological hardware and limited transmission speed of neuronal brainwave signals [48]. As a prevailing explanation, a primate's visual perception system tends to leverage intermediate visual processes to ***select*** a subset of the important sensory information before further intensive processing [49]. As a result, the system can spontaneously attend to the regions of target stimulus of a scene while ignoring the less-relevant parts [50, 51] such that enables effective scene analysis with constrained burden of signal processing [52]. This special neuronal behavior is so-called *focus of attention*.

By imitating the neuronal *focus of attention*, the (artificial) neural self-attention and self-gating mechanisms are introduced to realize effective yet efficient information *selection* in deep-learning-based visual recognition through adaptive neural feature re-weighting [53]. In the subsequent, we briefly introduce the basic neural self-attention and self-gating mechanisms as preliminary knowledge of our research in this dissertation.

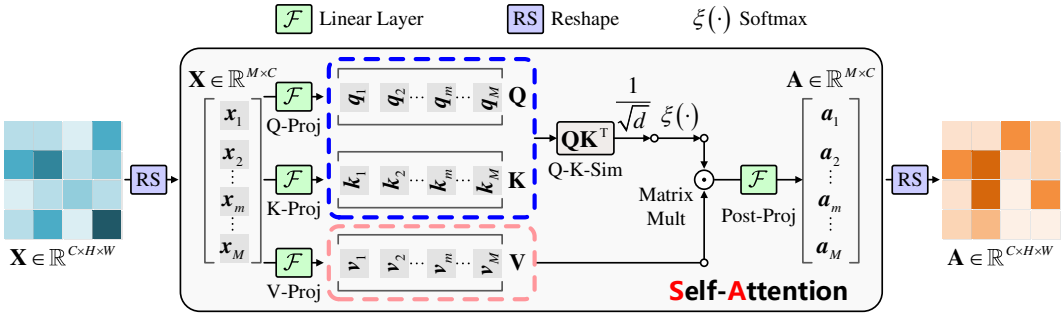


Figure 2.1: Operational illustration of the original spatial self-attention. $M = H \cdot W$ and $\mathbf{A}$ denotes the attended feature map.

2.1.1 Self-Attention

The neural *self-attention* model was first presented by Cheng *et al*. [34] to help relate different positions of a single structured sequence in an LSTM unit for modeling attentive transduction. Vaswani *et al*. [35] then extended the (multi-head) self-attention mechanism as a general building component in the MetaFormer architectural block to realize a dynamic and selective token mixing. Wang *et al*. [54] generalized the multi-head self-attention to visual recognition with $2D$ image inputs the first time. They applied self-attention as a plug-and-play module to learn fine-grained long-range feature dependencies on the spatial lattice. Dosovitskiy *et al*. [19] then suggested the highly scalable spatial window (self-)attention paradigm working on a partitioned feature map. This constrains the computational complexity of spatial self-attention on high-resolution $2D$ inputs, which underlie the current vision Transformers. Note that the (general) window attention is operationally equivalent to the vanilla spatial self-attention **w**ith **r**espect **t**o (**w.r.t.**) the processing on each individual image patch. Following we introduce the vanilla spatial self-attention (illustrated in Figure 2.1) as a preliminary knowledge.

Self-attention models the dot-product-based vector-wise relative feature similarities on the spatial lattice, produces the discriminative fine-grained weights by soft-maxing the relative feature similarities, and generates the context-aware attended features by dynamically re-integrating the re-weighted feature vectors. To realize such a self-attention process, for a given input feature map $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$, where C and $H \times W$ denote the number of channels and the spatial resolution, respectively, a query $\mathbf{Q}$, a key $\mathbf{K}$, and a value $\mathbf{V}$ are each converted from $\mathbf{X}$ through the linear projections $\mathcal{F}_Q$, $\mathcal{F}_K$, and $\mathcal{F}_V$, respectively. By reshaping the query, key, and value from the size $C \times H \times W$ to $G \times N \times D$, respectively, where $N = H \cdot W$, G denotes the number of heads, and $D = {}^{C}/_{G}$ denotes the length of each head, the overall self-attention (denoted by $\mathcal{A}$) is computed as:

$$\mathcal{A}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{D}}\right)\mathbf{V}, \tag{2.1}$$

where T denotes the transpose operator.

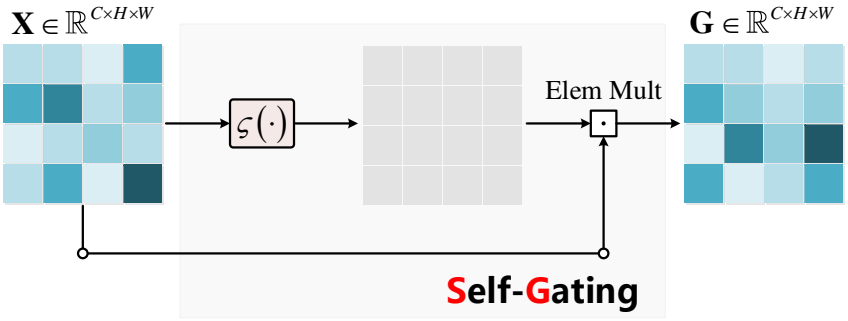


Figure 2.2: Operational illustration of a typical (soft) self-gating process. $\mathbf{G}$ denotes the gated feature map.

For clarity, we let $\boldsymbol{x}(h,w) \in \mathbb{R}^C$ denotes an arbitrary feature vector on $\mathbf{X}$, where $(h,w) \in \Omega_{H\times W}$. $\boldsymbol{q}_g(h,w), \boldsymbol{k}_g(h,w), \boldsymbol{v}_g(h,w) \in \mathbb{R}^D$ denote the query, key, and value vectors after the reshaping (here, g denotes the index of head), respectively. That is, a spatial self-attention models a discriminative weighted summation process where each attended feature vector at (h,w) is generated by a weighted-summation of the value vectors, where the weights are calculated as the relative query-key similarities (*i.e.*, scaled softmaxed query-key dot-products).

2.1.2 Self-Gating

In contrast to self-attention, the self-gating mechanism calculates feature weights based on the direct local cues and/or statistical feature responses of non-local cues with a non-linear curve that introduces targeted inductive biases for feature selection. Such a feature re-weighting process can be static yet more efficient than the counterpart within a self-attention.

For a given feature map $\mathbf{X} \in \mathbb{R}^{C\times H\times W}$, where $x_c(h,w) \in \mathbb{R}$ is an arbitrary (scalar-valued) feature unit on $\mathbf{X}$, a basic self-gating process (denoted by ϕ) of $x_c(h,w)$ can be described as:

$$\phi(x_c(h,w)) = \varsigma(x_c(h,w))\, x_c(h,w) \ , \tag{2.2}$$

where $\varsigma : \mathbb{R} \to \mathbb{R}$ defines the re-weighting function of ϕ. Typical ς induces a soft selection on the input (as illustrated in Figure Figure 2.2), which is continuous, smooth, (upper- and lower-)bounded, differentiable (at least first-order), and monotonically increasing (non-decreasing) on $\mathbb{R}$, *e.g.*,

$$\varsigma(x_c(h,w)) = \frac{e^{x_c(h,w)}}{e^{x_c(h,w)}+1} \ , \tag{2.3}$$

i.e., a Sigmoid function, and more alternatives can be applied (*e.g.*, $\varsigma(\cdot) = 0.5\,(1+\operatorname{erf}(\cdot/\sqrt{2}))$ [12] and $\varsigma(\cdot) = \tanh(\operatorname{softplus}(\cdot))$ [14]).

In particular, ς can induce a hard selection on $x_c(h, w)$ (as illustrated in Figure xxx) if it satisfies:

$$\varsigma(x) = \begin{cases} 0, & x_c(h, w) \leq \eta; \\ \varsigma, & x_c(h, w) > \eta, \end{cases} \tag{2.4}$$

where $\varsigma \in \mathbb{R}, \varsigma \neq 0$ (the most common case is $\varsigma = 1$) and $\eta \in \mathbb{R}$ denotes a given threshold for the selection.

2.2 Basic Methods for Multi-Criteria Decision-Making

Multi-**C**riteria **D**ecision-**M**aking (**MCDM**) is a sub-discipline of operational research. In a typical MCDM process, different (available) alternative candidates for a specific objective are comprehensively evaluated based on a given/defined set of criteria (*i.e.*, indicators) to make the decision or the conclusion of an analysis, where each criterion measures a corresponding available attribute of the alternative candidates that relate to the concerned objective. By generalizing the basic operations in a neuron to an MCDM process, we propose the new MCDM hypothesis for neural feature activation which originates our novel **IIEU** and **AdaShift** prototypes for learning improved self-gated neural activation.

In particular, three prevalent methods for MCDM, namely, **T**echnique for **O**rder **P**reference by **S**imilarity to **I**deal **S**olution (**TOPSIS**), **F**uzzy **C**omprehensive **E**valuation (**FCE**), and **G**rey **R**elational **A**nalysis (**GRA**) are introduced as the preliminary knowledge of our research as follows.

2.2.1 TOPSIS

The **T**echnique for **O**rder **P**reference by **S**imilarity to **I**deal **S**olution (**TOPSIS**) is a popular MCDM model, wildly applied in the analysis of economic, commercial, environmental data, which is built based on core idea that the chosen alternative candidate should have the (relative) shortest geometric distance from the positive ideal candidate (solution) while the longest geometric distance from the negative ideal candidate (solution), simultaneously [55]. The first TOPSIS method was originally introduced by Hwang and Yoon [56] and developed by Yoon [57] and Hwang *et al*. [58]. Based on these fundamental efforts, new derivatives with specified modifications have been proposed to address different real-world applications [59, 60, 61, 62, 63, 64, 65, 66, 67]. Following we introduce the basic TOPSIS process with the assumed simple preliminary settings.

Preliminaries.
We consider multi-criteria decision analysis on N different alternative candidates with C concerned criteria for the comprehensive assessments, where $N > 2$ and $C > 2$. For the n-th candidate, its original measured score on the c-th criterion, *i.e.* the intersection of each candidate and criteria, is denoted by $x_{n,c} \in \mathbb{R}$.

Process.

Step1. Based on the preliminary settings, first create an evaluation matrix $\boldsymbol{X} = (x_{n,c})_{N\times C}$ consisting of the measured scores of the N alternative candidates on the C relevant criteria.

Step2. To ensure the numerical comparability between different criteria, for the total C criteria, their corresponding scores of the N candidates are then normalized through intra-criterion normalization. This converts the evaluation matrix $\boldsymbol{X}$ to the normalized evaluation matrix $\boldsymbol{R} = (r_{n,c})_{N\times C}$, where each $r_{n,c}$ is calculated by

$$r_{n,c} = \frac{x_{n,c}}{\sqrt{\sum_{i=1}^{N} x_{i,c}^2}}\,, n = 1, 2, \ldots, N\,, c = 1, 2, \ldots, C\,. \tag{2.5}$$

Step3. By taking into account the differences of significance between different criteria (for the addressed application), the weighted normalized decision matrix $\boldsymbol{Z} = (z_{n,c})_{N\times C}$ is calculated by re-weighting the C criteria of the normalized evaluation matrix $\boldsymbol{R}$, respectively. That is,

$$z_{n,c} = \lambda_c \cdot r_{n,c}\,, n = 1, 2, \ldots, N\,, c = 1, 2, \ldots, C\,, \tag{2.6}$$

where $\lambda_c \mid \lambda_c \in \mathbb{R}, \lambda_c \geq 0, \exists \lambda_c \neq 0$ denotes the normalized weight of the c-th criterion, *i.e.*,

$$\lambda_c = \frac{\kappa_c}{\sum_{j=1}^{C} \kappa_j}\,, j = 1, 2, \ldots, C\,, \tag{2.7}$$

where $\kappa_c \mid \kappa_c \in \mathbb{R}, \kappa_c \geq 0, \exists \kappa_c \neq 0$ denotes the original bounded weight assigned to the c-th criterion. That is, the normalized weights $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \ldots, \lambda_C] \in \mathbb{R}^C$ satisfies

$$\sum_{j=1}^{C} \lambda_j = 1\,, j = 1, 2, \ldots, C\,. \tag{2.8}$$

Step4. Determine the positive ideal candidate $\boldsymbol{z}^+ = \left[z_1^+, z_2^+, \ldots, z_C^+\right]$ and negative ideal candidate $\boldsymbol{z}^- = \left[z_1^-, z_2^-, \ldots, z_C^-\right]$, where each

$$z_c^+ = \max\left\{z_{n,c} \mid n = 1, 2, \ldots, N\right\}|_{c\in\mathbb{I}^+} \vee \min\left\{z_{n,c} \mid n = 1, 2, \ldots, N\right\}|_{c\in\mathbb{I}^-}\,, \tag{2.9}$$

$$z_c^- = \min\left\{z_{n,c} \mid n = 1, 2, \ldots, N\right\}|_{c\in\mathbb{I}^+} \vee \max\left\{z_{n,c} \mid n = 1, 2, \ldots, N\right\}|_{c\in\mathbb{I}^-}\,, \tag{2.10}$$

where the set $\mathbb{I}^+$ is associated with the criteria that have a *positive* impact (*i.e.*, the so-called *benefit criteria*), and the set $\mathbb{I}^-$ is associated with the criteria that have a *negative* impact (*i.e.*, the so-called *cost criteria*). "$\vee$" denotes logical "OR." Note that $\boldsymbol{z}^+ \neq \boldsymbol{z}^-$ is assumed.

Step5. Calculate the $\mathcal{L}_2$-distances from each alternative candidate-n (represented by $\boldsymbol{z}_n$) to the positive ideal candidate $\boldsymbol{z}^+$ and the negative ideal candidate $\boldsymbol{z}^-$, respectively, *i.e.*,

$$d_n^+ = \left\|\boldsymbol{z}_n - \boldsymbol{z}^+\right\|\,, \quad d_n^- = \left\|\boldsymbol{z}_n - \boldsymbol{z}^-\right\|\,, \tag{2.11}$$

where d_n^+ and d_n^- denote the $\mathcal{L}_2$-distances from $\boldsymbol{z}_n$ to $\boldsymbol{z}^+$ and $\boldsymbol{z}^-$, respectively.

Step6. Score each alternative candidate-n based on the measured positive and negative distances d_n^+ and d_n^-, *i.e.*,

$$\mathrm{score}_n = \frac{d_n^-}{d_n^- + d_n^+}\,, \qquad \mathrm{score}_n \in [0,1]\ . \tag{2.12}$$

In particular, $\mathrm{score}_n = 1$, *i.e.*, $d_n^+ = 0$ if and only if the candidate-n equals to the positive ideal candidate. In contrast, $\mathrm{score}_n = 0$, *i.e.*, $d_n^- = 0$ if and only if the candidate-n equals to the negative ideal candidate.

Step7. Finally, rank the N alternative candidates according to their comprehensive scores. Generally, the chosen candidate (*i.e.*, decision) is the one that has the relatively highest comprehensive score.

Entropy-based weight calculation for TOPSIS.
In **Step 3**, a popular approach to calculate the original weight (vector) $\boldsymbol{\kappa}$ is the Entropy Method (EM) [61]. The EM computes weights for different criteria based on the information entropy of the scores of the alternative candidates on the criteria, where a criterion with a relatively small entropy is assigned with a relatively large weight as indicating higher commonality among various candidates. Below is the calculation of EM.

For a given normalized evaluation matrix $\boldsymbol{R} = (r_{n,c})_{N\times C}$, the information entropy of the c-th criterion, E_c, is calculated by:

$$E_c = -\frac{1}{\ln(N)}\sum_{i=1}^{N} r_{i,c}\ln r_{i,c}\,. \tag{2.13}$$

Then, the entropy weight of the c-th criterion is defined by:

$$\kappa_c = 1 - E_c\,, \tag{2.14}$$

and the normalization on the entropy weights $\boldsymbol{\kappa}$, which produces the normalized weights $\boldsymbol{\lambda}$, follows Equation (2.7).

2.2.2 Fuzzy Comprehensive Evaluation

The MCDM method **F**uzzy **C**omprehensive **E**valuation (**FCE**) is an application of the fuzzy set [68, 69, 70]. It makes comprehensive assessments of the given alternative candidates in a fuzzy decision condition with multiple concerned criteria. The basic goal of FCE is to convert the original qualitative evaluation into a quantitative evaluation based on the membership function, where the comprehensive scores of the candidates are calculated with fuzzy operators. FCE makes a synthetic assessment of each alternative candidate by leveraging

the fuzzy transformation based on the maximum membership or weighted-average principles from a holistic perspective of the relevant indicators [71]. Following we introduce the process of basic FCE on an individual alternative candidate for evaluation.

Process.

Step1. First provide a set of criteria $\mathbb{I} = \{I_1, I_2, \ldots, I_C\}$ for decision making, each of which reflects the performance of the target candidates at a concerned aspect.

Step2. Provide a quantized grade set $\Xi = \{\xi_1, \xi_2, \ldots, \xi_K\}$. Each grade ξ_k denotes a certain quantized level of evaluation for a criterion.

Step3. Establish the evaluation matrix $\boldsymbol{X} = (x_{c,k})_{C\times K}$ consisting of all the quantized single-factor evaluations of the evaluated alternative candidate on the C criteria with the total K levels of grade. That is, for a row $\boldsymbol{x}_c = [x_{c,1}, x_{c,2}, \ldots, x_{c,K}]$ on $\boldsymbol{X}$, each value $x_{c,k}$ represents a membership degree of the grade level ξ_k to the criterion I_c. Then, the normalized evaluation matrix $\boldsymbol{R} = (r_{c,k})_{C\times K}$ is obtained by letting

$$r_{c,k} = \frac{x_{c,k}}{\sum_{j=1}^{K} x_{c,j}}, c = 1, 2, \ldots, C\,, \tag{2.15}$$

i.e., $\sum_{j=1}^{K} r_{c,j} = 1$. Note that for a criterion I_c, its corresponding membership values of different grades $\boldsymbol{x}_c$ can be determined by the statistical results (*e.g.*, raw average) of multiple evaluations.

Step4. Determine the normalized weights $\boldsymbol{\lambda} \in \mathbb{R}^C$ for the C criteria. The operations can be described by Equations (2.6) to (2.8), *i.e.*, the **Step3** of TOPSIS, where the entropy method (*i.e.*, Equations (2.13) and (2.14)) is applicable.

Step5. Generating the synthetic scores $\boldsymbol{B} = [b_1, b_2, \ldots, b_K]$ for the alternative candidate based on $\boldsymbol{R}$. The synthetic scores $\boldsymbol{B}$ are calculated by multiplying the weights of criteria $\boldsymbol{\lambda}$ (a vector) with the normalized evaluation matrix $\boldsymbol{R}$. That is,

$$\boldsymbol{B} = \boldsymbol{\lambda} \circ \boldsymbol{R}\,, \tag{2.16}$$

where "$\circ$" denotes an optional kind of fuzzy operator to combine the two factors. The commonly applied fuzzy operators for "$\circ$" operation are described as follows.

Operator-1: principal factor determined.

$$\boldsymbol{M}\,(\wedge, \vee) : b_k = \max\nolimits_{c=1}^{C} \{\min\{\lambda_c, r_{c,k}\}\}\ . \tag{2.17}$$

Here, "$\wedge$" and "$\vee$" represent Zadeh's [72] logical operator defined for fuzzy set computations, where

$$\lambda_c \wedge r_{c,k} = \min\{\lambda_c, r_{c,k}\} , \tag{2.18}$$

$$\lambda_c \vee r_{c,k} = \max\{\lambda_c, r_{c,k}\} , \tag{2.19}$$

respectively.

Operator-2: principal factor prominent.

$$\boldsymbol{M}(\cdot, \vee) : b_k = \max_{c=1}^{C} \{\lambda_c \cdot r_{c,k}\} . \tag{2.20}$$

Operator-3: weighted average.

$$\boldsymbol{M}(\cdot, \oplus) : b_k = \sum_{i=1}^{C} \lambda_c \cdot r_{c,k} . \tag{2.21}$$

Operator-4: rectified average.

$$\boldsymbol{M}(\wedge, \oplus) : b_k = \sum_{i=1}^{C} \{\min\{\lambda_c, r_{c,k}\}\} . \tag{2.22}$$

Step6. Assess the alternative candidate by the final synthetic score score based on $\boldsymbol{B}$. The score is calculated as:

$$\mathrm{score} = \max_{k=1}^{K} \{B_k\} , \tag{2.23}$$

if using the maximum membership principle and calculated as:

$$\mathrm{score} = \frac{\sum_{j=1}^{K} B_j \cdot \xi_j}{\sum_{j=1}^{K} B_j} , \tag{2.24}$$

if applying the weighted average principle.

Step7. Rank the N alternative candidates according to their comprehensive scores and choose the one that has the relatively highest comprehensive score.

2.2.3 Grey Relational Analysis

The **G**rey **R**elational **A**nalysis (**GRA**) [73] is a popular grey-system-based model for MCDM in various real-world applications [74, 75, 76]. As the key idea, GRA defines two virtual ideal cases as the references for evaluation, where the case of *black* refers to *zero-information* and the case of *white*, in contrast, refers to *perfect information*. The real-world cases that contain *partial information* are then assumed between the virtual negative and positive ideal cases. Based on these assumptions, GRA comprehensively scores each real

alternative candidate based on its distance from the virtual positive/negative ideal candidate. Following we introduce the scoring process of the basic GRA (with the assumed virtual positive candidate) by adopting the preliminary settings of TOPSIS.

Process.

Step1. Assume (1) $\boldsymbol{x}_n = [x_{n,1}, x_{n,2}, \ldots, x_{n,C}] \in \mathbb{R}^C$ denotes the original evaluation scores of the n-th alternative candidate on the C criteria, where $n = 1, 2, \ldots, N$; (2) $\boldsymbol{x}_0 = [x_{0,1}, x_{0,2}, \ldots, x_{0,C}] \in \mathbb{R}^C$ denotes the reference ideal candidate for evaluation.

Step2. Convert each $\boldsymbol{x}_n$ to the corresponding normalized evaluation scores $\boldsymbol{r}_n \in \mathbb{R}^C$ by adopting Equation (2.5).

Step3. Determine the normalized weights $\boldsymbol{\lambda} \in \mathbb{R}^C$ for the C criteria by adopting Equations (2.6) to (2.8), where the entropy method (*i.e.*, Equations (2.13) and (2.14)) is applicable.

Step4. Compute the **G**rey **R**elational **C**oefficients (**GRC**) by

$$\zeta_{n,c} = \frac{\min_{i=1}^{N} \min_{j=1}^{C} |x_{0,j} - x_{i,j}| + \vartheta_{n,c} \max_{i=1}^{N} \max_{j=1}^{C} |x_{0,j} - x_{i,j}|}{|x_{0,c} - x_{n,c}| + \vartheta_{n,c} \max_{i=1}^{N} \max_{j=1}^{C} |x_{0,j} - x_{i,j}|}, \tag{2.25}$$

where $\zeta_{n,c} \in \mathbb{R}$ and $\vartheta_{n,c} \in (0, 1]$ denote the GRC and the **D**ynamic **D**istinguishing **C**oefficient (**DDC**) of the candidate-n on the criterion-c, respectively.

Step5. Compute the comprehensive scores by

$$\mathrm{score}_n = \sum_{j=1}^{C} \lambda_j \cdot \zeta_{n,j}, \tag{2.26}$$

where score_n is the final comprehensive score of the n-th alternative candidate.

Step7. Rank the N alternative candidates according to their comprehensive scores and choose the one that has the relatively highest comprehensive score.

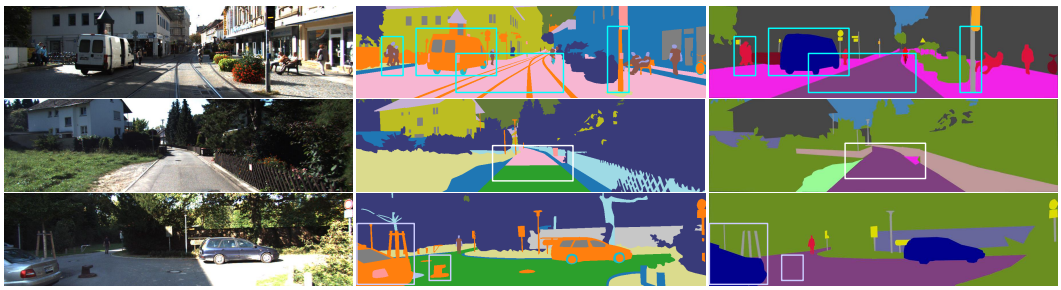
Chapter 3

RGB Road Scene Material Segmentation

3.1 Background

Recognition of materials is critical for real-world computer vision applications, since materials, what objects are made of, inform the informative physical attributes of semantic objects and regions in a scene that are unreachable by attending only to object categories. Planning an action for an iron-made board would be different from the way for a wood board, and informative visual cues would be favorable. The significance of dense per-pixel RGB material recognition can be even higher for road scenes, especially for the safety and effective driving planning of autonomous vehicles that navigate in diverse traffic environments. Despite the prospective contributions to safe navigation and past efforts on object- and image-level material recognition, there have been limited investigations on visual road scene material understanding from regular color images that are easy to access.

Per-pixel RGB material recognition (*i.e.*, material segmentation, in contrast to semantic segmentation) would favor driving assistance systems and autonomous driving significantly. For example, the roads paved with *asphalt* and *brick* mean very different for intelligent speed planning. Distinguishing a *plastic* model that resembles a pedestrian in shape from a real pedestrian may help to anticipate the driving movements. RGB road scene material segmentation cannot be regarded as another semantic segmentation task with a different set of labels, as they are significantly different in the level of challenges. As an example, roads can be built of “asphalt,” “concrete,” or even “brick,” but they are indiscernible from shape.



(a) Image (b) Material (c) Semantics

Figure 3.1: Materials versus Semantics. Top: A single *semantic* object may be composed of multiple *material* ingredients and different *semantic* objects possibly contain the same *material* ingredient. Middle: The object "Road" can be built of "asphalt," "concrete," or even "brick," while indiscernible from shapes. Bottom: A metal-made "obstacle" that is unclear in the semantic annotations, possibly poses a driving hazard. ©2022 Springer Nature [1]

This challenge can be intensified by the fact that the surface of an individual object can be composed of various material ingredients, *e.g.*, a car composed of "metal," "glass," "rubber, vinyl," and "plastic." In contrast, characteristic shapes often serve as a type of primary cues for discerning semantic objects. Figure 3.1 illustrates the key differences between RGB road scene material and semantic segmentation.

Through careful analysis on images from KITTI-Materials [1], the first benchmark dataset focusing on RGB RMS the emerging avenue of visual recognition research, we identify the effective extraction and fusion of local texture cues and image context of materials as the key to generating accurate representations of material appearance in road scenes. More specifically, the signature textures of materials inform discriminative cues for their identification. The material textures, however, may change drastically with occlusion and scale, *i.e.* distance from viewpoint, which hinders the learning of robust neural representations. In object segmentation, the incorporation of effective structural dependencies contributes to refining the texture cues by emphasizing representations relatively robust to the scale and occlusion variations. Whereas, the direct generalization of this knowledge to materials likely faces significant challenges owing to the weak and delicate shape cues of typical road scene material regions. To address this problem, we suppose that the effective use of the scale-aware fine-grained context of materials, which helps to highlight the levels and scales of features that attend to the corresponding material textures, is even critical to learning discriminative joint neural representations for RGB RMS. Despite its significance for RGB RMS, the adaptive fusion of effectively mixing scale-aware context and texture cues remains a challenging task, especially for Transformer-induced features, where the popular/SOTA dense segmentation heads [77, 3, 78] show limited performances (we introduce this phenomenon in Section 3.5.3 with experimental results and a relevant problem is also reported by Xie [4] in object segmentation). This suggests that we need a new tailored model to realize effective context-aware multi-scale feature fusion for accurate RGB RMS.

Our above intuition of the important properties of material appearance (in color images) originates the new **RMSNet** framework, constructed on the novel *self-attention*-based **SAMixer** model that enables effective context-aware multi-scale feature fusion with high efficiency, by addressing the two critical problems for MSA-induced feature fusion, *i.e.*, *imbalanced partial query-key similarity measure* and *high complexity*. As its core idea, SAMixer generalizes the original MSA of quadratic complexity that integrates feature vectors on a spatial lattice to a specialized efficient feature aggregator that mixes features of long-range contextual cues and local texture cues at each aligned pixel location by introducing the new balanced **Q**uery-**K**ey-**Sim**ilarity measure. The balanced Q-K-Sim measure is computed with a container feature generated by aggregating all the multi-level feature maps (*i.e.*, the inputs to the SAMixer). It models comprehensive Q-K relationships in each individual weight calculation for the key with the help of the container feature, such that a balanced Q-K similarity can be computed with a single vector-matrix multiplication. This enables our SAMixer to realize the first highly efficient many-to-one feature fusion through the pure *Q-K-V interactive attention* process (to the best of our knowledge) with only linear complexity (specifically, $O(N)$ complexity, where N denotes the number of input feature maps). To further exploit the local details of features, we also present the **B**ottleneck **L**ocal **S**tatistics **E**ncoding-**D**ecoding (BLSED) strategy as an embedded enhancing scheme for the balanced Q-K-Sim measure in SAMixer. It encodes local statistics within each of the small regular partitions on the objective feature maps so that the regional context cues of the neighborhoods can be extracted and exploited to augment MSA-based feature fusion. It then decodes the local statistics after the MSA process to produce a series of feature patches and merges each feature patch with the attended feature at the aligned positions to incorporate rich local details and generate the high-resolution attended feature map.

To obtain a set of effective multi-scale features for fusion (*i.e.*, inputs of SAMixer), RMSNet leverages efficient hierarchical encoder(s) [4, 21, 22] equipped with positional **F**eed-**F**orward-**N**etwork (**FFN**) to extract local texture features and long-range context from multi-level hierarchies.

We believe our work of RGB RMS, the emerging avenue of research, can contribute to richer visual understanding, particularly of road scenes, for safer driving. RMSNet will serve as a sound baseline framework for this important task.

The contributions of this work are 3-fold:

1. we investigate the unexplored yet significant visual recognition problem for general scene understanding, RGB road scene material segmentation, based on deep learning techniques;
2. we propose RMSNet, a new baseline deep learning framework for improving RGB RMS, built on the novel context-aware multi-level multi-scale fusion model, SAMixer, for enhancing neural features, especially for Transformer-induced features, which to the best of our knowledge, realizes the first highly efficient many-to-one feature fusion

through the pure Q-K-V attention process for dense per-pixel recognition with linear complexity;

3. and we validate our RMSNet (SAMixer) through extensive quantitative analysis and ablation studies on the RGB RMS benchmark, *i.e.*, KITTI-Materials. Experimental comparisons with various relevant RGB material segmentation and road scene semantic segmentation methods clearly demonstrate the effectiveness. Moreover, the auxiliary evaluations on the relevant datasets MCubeS and images from Cityscapes verify the generalizability of our RMSNet (SAMixer) for RGB RMS on the realistic driving view images.

3.2 Related Work

Bell *et al*. [79] demonstrated material segmentation with a fully convolutional network cascaded with a fully-connected CRF [80, 81, 82], which is essentially semantic segmentation with a new set of annotations of material classes mainly consisting of architectural images. Schwartz and Nishino proposed the use of material properties as an intermediate representation for dense material recognition, which was free from modeling explicit shape features [83, 84, 85]. Subsequently, they [32] introduced the integration of global context cues through the form of semantic segmentation and place recognition. They demonstrated its practical application in material segmentation by constructing a material dataset consisting of local image patches sourced from ImageNet [86] and COCO dataset [23]. Relevantly, Zhang *et al*. suggested reflectance hashing for efficient and accurate material recognition [87].

Xue *et al*. [88] investigated the advantage of differential angular imaging for material recognition on the GTOS dataset constructed by photographs of ground surface material that are captured as top-down fronto-parallel images. Zhang *et al*. [89] introduced the DeepTEN network based on orderless texture encoding [90]. Later, Xue *et al*. [91] improves the model in [88] by incorporating texture encoding These methods, however, focus on image-wise material recognition, where [88] and [88] applied differential angular imaging.

Demir *et al*. [92] investigated road and building extraction on DeepGlobe dataset of satellite images. Similar efforts have also been taken by Purri *et al*. [93]. They proposed reflectance residual encoding for material segmentation from satellite images [94]. More recently, Xue *et al*. [95] presented AngLNet that leverages per-pixel angular luminance from multiple views. Material segmentation on road scenes is distinct from these tasks of bird-eye-view material segmentation as scale variation, *i.e*. distance variation from viewpoint, is inevitable due to the dynamic driving perspective.

The RGB road scene *semantic* segmentation stands as a popular research field, offering us valuable insights for model design. A series of works [96, 97, 98, 3, 99, 100, 101, 102, 103, 104, 4] have addressed road scene *semantic* segmentation on Cityscapes [105] dataset

that consists of realistic driving view images captured by car-mounted cameras. In contrast, RGB road scene *material* segmentation has not been intensively investigated.

A related work to our RGB RMS is MCubeS [106] multimodal material segmentation. Liang *et al*.introduced MCubeS dataset and presented multimodal material segmentation of city scenes on MCubeS dataset by leveraging the combined information of RGB, NIR, and polarization images. They proposed MCubeSNet, a new CNN-based material segmentation model, built on DeepLabv3+ [3], equipped with a newly derived region-guided filter selection (RGFS) layer to exploit different combinations of imaging modalities guided by semantic segmentation. In contrast, our focus is realizing effective RGB road scene material segmentation.

In particular, in contrast to the general viewpoint and scene material segmentation, our goal is to realize inference-efficient RGB RMS and address the scale variation, the prevalent challenge in road scenes, which makes this emerging visual recognition problem particularly challenging. High inference efficiency is important for RGB RMS as a basic intention of RGB RMS is ultimately facilitating real-time perception systems for autonomous driving/driving assistance. Compared to static general scenes, we find RGB RMS also particularly challenging since the viewpoint is highly dynamic and in a scene structure with wide depth ranges. These exacerbate the complex scale variations, where the signature material textures can dramatically vary with their changing distances from the viewpoint, but they need to be identified as the same material categories. In contrast, images of general scenes can exhibit relatively gentle and simple variations of scale. Moreover, moving objects, *e.g*., vehicles and pedestrians, important components of road scenes, also likely intensify the complexity of scale variations due to their complicated movements. Therefore, we introduce RMSNet with SAMixer as a tailored model for RGB RMS, which leverages these appearance variations of materials in road scenes

Vision Transformers leverage **M**ulti-head **S**elf-**A**ttention (**MSA**) [35] to model long-range visual cues [54, 107, 108]. Original MSA to $2D$ spatial features, however, incurs excessive computational burden. Ramachandran *et al*. [103] modified the Transformer to work on a fixed region and added explicit positional biases. Wang *et al*. [104] introduced the stand-alone Axial-MSA which processes feature maps along the height- and the width-axis, respectively, to balance computational cost and accuracy for segmenting semantic objects. Zhang *et al*. [109] demonstrated that the co-occurrence of semantics, including object categories, exhibits non-local dependencies.

ViT [19] calculates MSA within each non-overlapping image zone to achieve a speed-accuracy tradeoff for object recognition. PVT [110] suggested the first pyramid vision Transformer architecture and showed its potential for dense prediction tasks. Liu *et al*. [26] applied MSA within fine-grained shifted windows and model inter-window relationships to augment local details. LeViT [111] and TNT [112] also proposed enhanced window-MSA by infusing extra local cues. Pure window-MSA, however, is still computationally expensive for high-resolution features and likely neutralizes the local details if applied to

the application where the resolutions are changeable.

SegFormer [4] built a hierarchical Transformer encoder with an efficient MSA where the keys and values with reduced resolution were computed from condensed features with convolutions. It also introduced an effective All-MLP decoder that jointly uses multiple levels of the encoded features. A related idea was suggested in CvT [20], where the Q-K-V projections were realized by convolutions. Most recently, Ding *et al*. [21] suggested a hierarchical dual attention vision Transformer, namely, DaViT, which learns non-local image context within both spatial and channel token spaces with sequential MSA layers. Liu *et al*. [22] incorporated the block design of MetaFormer [25] into a ConvNet and introduced ConvNeXt as a new SOTA hierarchical backbone that matched SOTA vision Transformers with high efficiency. Our RMSNet employs efficient yet effective hierarchical backbone(s) to encode multi-level features (four levels by default) and leverages the novel SAMixer model to fuse multi-level features of local textures and long-range contextual cues to generate robust representations for road scene material segmentation.

Past works have explored multi-scale feature learning in object recognition and semantic segmentation. Chen *et al*. [113] plugged a spatial attention layer into the bottom of a two-branch network to learn weights for features at different scales. SKNet [114] expanded SE-Net [18] to aggregate multi-scale features. The Deeplab family [115, 3, 99, 98] used atrous spatial pyramid pooling to learn scale-invariance through global statistics and a set of convolutions with different dilations. Xiao *et al*. [77] proposed the Unified Perceptual Parsing Network (UperNet) which integrates the Feature Pyramid Network (FPN) [116] and Pyramid Pooling Module [96] in the segmentation head to extend the effective receptive field of CNN and learn multi-level feature representation with a top-down architecture and lateral connections.

More recently, Deformable DETR [37] for object detection introduced feature pyramids learned with the help of MSAs, and the final integration of multi-scale features relies on extra convolutional operations outside the MSAs. HRViT [117] shared multi-scale information across the layers of a vision Transformer encoder through direction-decomposed local MSAs (termed as HRViTAttns) to avoid memory explosion. However, HRViTAttn is designed for N-to-N (*i.e*., N inputs and N outputs) inter-layer multi-scale information exchange, instead of a complete N-to-1 feature fusion. In contrast, we introduce SAMixer, a novel efficient yet effective MSA-based model to enhance multi-level scale-aware feature learning and fusion, which realizes complete N-to-1 multi-resolution multi-level feature fusion within the Q-K-V self-attention mechanism. Equipped with SAMixer, our RMSNet selectively activates meaningful features of local textures and non-local contextual interactions to form discriminative representations to achieve dense road scene material segmentation.

Table 3.1: Per-class pixel statistics for each scene in KITTI-Materials dataset. (1) "Scn ID" and "Imgs" denote "Scene ID" and "Images," respectively; "road mk," "fab, lthr," "rubr, vl," "cob," and "hum bd" denote "road marking," "fabric, leather," "rubber, vinyl," "cobblestone," and "human body," respectively. (2) Note that scene-0926095 includes an invalid pixel. "Trn-1, -2" and "Tst-1, -2" denote training and test sets of Split-1 and -2, respectively.

Scn ID	Imgs	asphalt	concrete	metal	road mk	fab, lthr	glass	plaster	plastic	rubr, vl	sand	gravel	ceramic	cob	brick	grass	wood	leaf	water	hum bd	sky	Pixels
0926002	1	83K	58K	13K	4125	17K	2314	0	0	3656	0	0	1018	0	20K	8919	566	137K	0	1035	39K	389K
0926019	50	2575K	501K	391K	23K	794	38K	0	3165	4850	0	0	0	0	207K	5579K	71K	9375K	308	95	686K	19M
0926039	50	2478K	472K	3039K	0	5297	1186K	6814K	85K	128K	0	0	411K	1158K	643K	375	16K	2107K	0	1954	911K	19M
0926048	5	0	162K	326K	0	0	258K	768K	8725	11K	0	0	9232	0	274K	5945	905	29K	0	0	93K	1946K
0926056	50	4340K	1237K	1517K	298K	11K	369K	188K	24K	26K	0	22K	114K	0	149K	2025K	723K	6934K	0	3170	1475K	19M
0926059	50	2635K	744K	2715K	341K	18K	821K	3175K	107K	115K	0	0	264K	1390K	701K	1481K	147K	3731K	0	4795	1066K	19M
0926064	50	2228K	4846K	2390K	7939	22K	745K	0	65K	88K	0	0	363K	1175	707K	44K	117K	7234K	714	7907	590K	19M
0926070	50	3251K	1576K	595K	264K	2611	87K	477K	25K	11K	0	0	75K	0	34K	4441K	416K	7144K	0	1489	1054K	19M
0926079	20	947K	316K	357K	0	0	97K	888K	4578	3582	0	0	39K	676K	0	343K	81K	3924K	0	0	106K	7782K
0926084	50	4444K	3761K	2519K	263K	13K	702K	0	55K	111K	0	0	67K	1190K	73K	576K	441K	4679K	0	3659	557K	19M
0926086	50	2203K	1478K	3582K	7604	25K	87K	2417K	119K	17K	0	1797	238K	508K	376K	1193K	253K	5785K	0	3874	1163K	19M
0926091	50	0	3967K	2603K	68K	519K	1260K	2399K	104K	139K	6789	0	298K	4781K	298K	41K	59K	2366K	0	80044	468K	19M
0926095	50	2802K	500K	2808K	62K	89K	1115K	4797K	59K	120K	5604	0	155K	1584K	2589K	124K	281K	1930K	0	31440	404K	19M
0926117	50	2808K	1270K	2641K	4468	57K	771K	0	54K	97K	25K	0	3580	643K	556K	1953K	544K	7699K	0	4311	326K	19M
0928037	15	366K	311K	794K	1668	192K	81K	125K	5065	63K	0	0	1282	1254K	217K	37K	126K	2107K	0	47515	108K	5837K
0928045	9	374	9552	154K	0	85K	415K	0	633	916	0	0	0	782K	521K	82K	511K	894K	0	11971	34K	3502K
0929004	50	2689K	602K	1220K	251K	0	487K	0	33K	54K	0	0	0	0	0	3400K	376K	9648K	0	0	696K	19M
0930016	50	4053K	638K	854K	593K	0	50K	190K	15K	11K	0	3497	175K	24K	1048K	1608K	74K	8762K	0	0	1356K	19M
0930020	50	2487K	1751K	3302K	20K	26K	177K	486K	30K	70K	269K	0	389K	1202K	48K	2866K	455K	3489K	0	4965	2385K	19M
0930033	50	2912K	168K	287K	103K	9962	12K	117K	2083	5772	387	0	61K	847K	5901	2499K	90K	10562K	0	1541	1772K	19M
0930034	50	742K	756K	373K	1545	0	45K	845K	32K	7065	17K	0	215K	1313K	275K	2365K	795K	10558K	0	0	1116K	19M
1003034	50	3951K	777K	660K	166K	0	89K	95K	12K	17K	0	36K	13842	1377K	40K	2570K	594K	8145K	0	0	912K	19M
1003042	50	3957K	719K	1585K	455K	0	61K	9931	6069	15K	0	0	0	0	0	2646K	2508	7321K	0	0	2677K	19M
1003047	50	3902K	4854	2488K	184K	0	481K	0	71K	134K	0	0	0	0	0	670K	3498	8398K	0	0	3120K	19M
Total	1K	56M	27M	37M	3121K	1092K	9437K	24M	920K	1253K	323K	63K	2891K	19M	8782K	37M	6178K	133M	1022	210K	23M	389M
Trn-1	800	46M	24M	30M	2904K	1066K	8786K	20530K	695K	1090K	306K	61K	2438K	16910K	7924K	26751M	5055K	99M	714	206K	17M	311M
Tst-1	200	9422K	2739K	6833K	217K	26K	651K	3262K	225K	163K	17K	1797	453K	1820K	859K	9807M	1123K	34M	308	3969	6085K	78M
Trn-2	800	44M	20M	29942K	2206K	981K	7040K	19M	748K	980K	318K	60K	2198K	17M	4439K	31M	5330K	105M	308	170K	20M	311M
Tst-2	200	12M	6587K	7271K	914K	111K	2397K	4987K	172K	273K	5604	3497	693K	1609K	435K	5175K	848K	28M	714	39K	3046K	78M

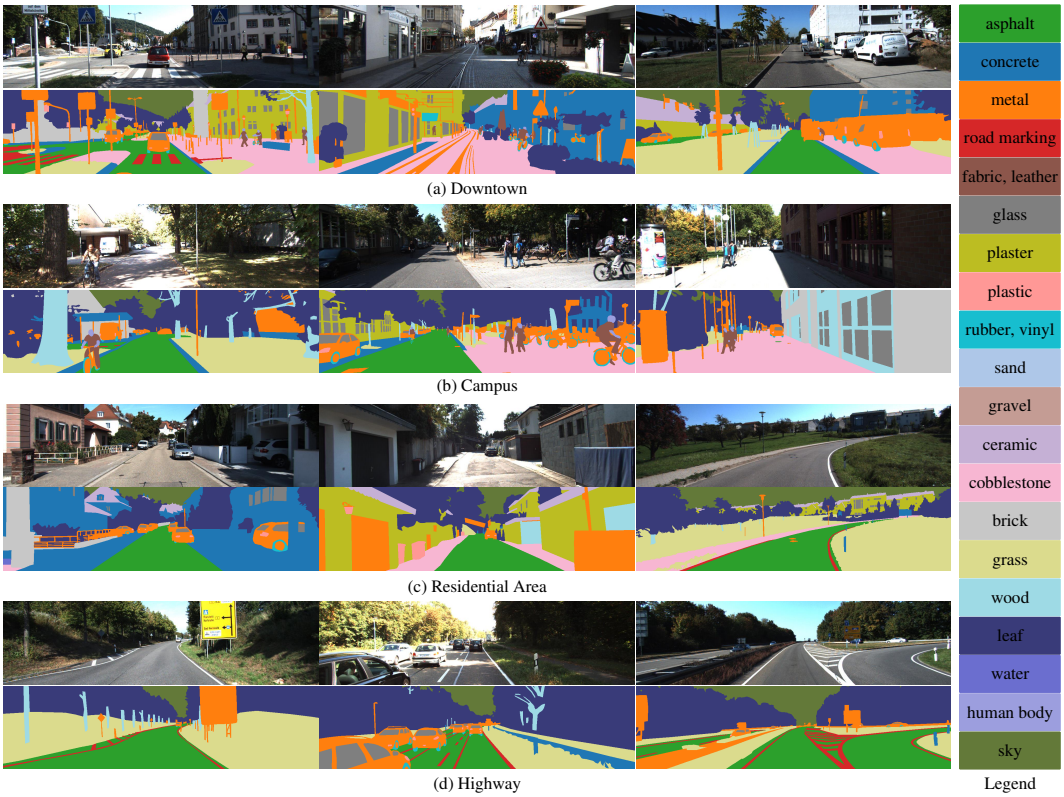


Figure 3.2: Example images and their corresponding material annotations from KITTI-materials dataset. From top to bottom are examples for "downtown," "campus," "residential area," and "highway," respectively. Different from road scene objects, materials have no signature shapes but show complex spatial distributions (*i.e.*, fragmented). Different objects may contain the same materials, and a single object can also have multiple regions of different materials. ©2022 Springer Nature [1]

3.3 Preliminary: KITTI-Materials Dataset

KITTI-Materials is the first comprehensive RGB road scene material segmentation dataset. The images composing KITTI-Materials are sourced from the raw data of KITTI benchmark suite [118]. Note that the images used in KITTI-Materials do not have the original semantic segmentation annotations. KITTI-Materials consists of 1000 images covering 24 different driving scenes including downtown, campus, residential area, highway, and other city/suburban landscapes captured from a car-mounted camera.

KITTI-Materials provides high-density per-pixel material annotations of 20 categories. All annotations of ground-truths are 320×1216 in resolution with the raw images center-cropped to this size beforehand. Figure 3.2 shows examples of the various types of road scene RGB images with their corresponding material annotations. As demonstrated in the visual examples, road scene materials lack signature shapes and exhibit complex spatial distributions.

Naturally reflecting the realistic driving environments, KITTI-Materials shows a strong

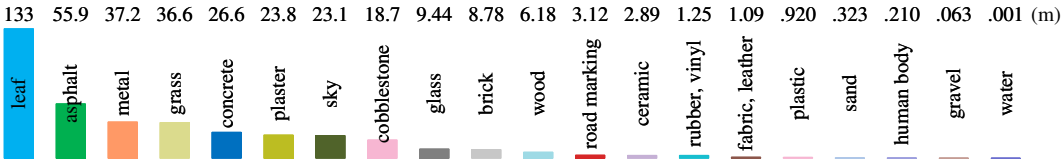


Figure 3.3: Per-class pixel statistics (in "millions") of KITTI-Materials. Pixel labels show a clear long-tail distribution of material categories. ©2022 Springer Nature [1]

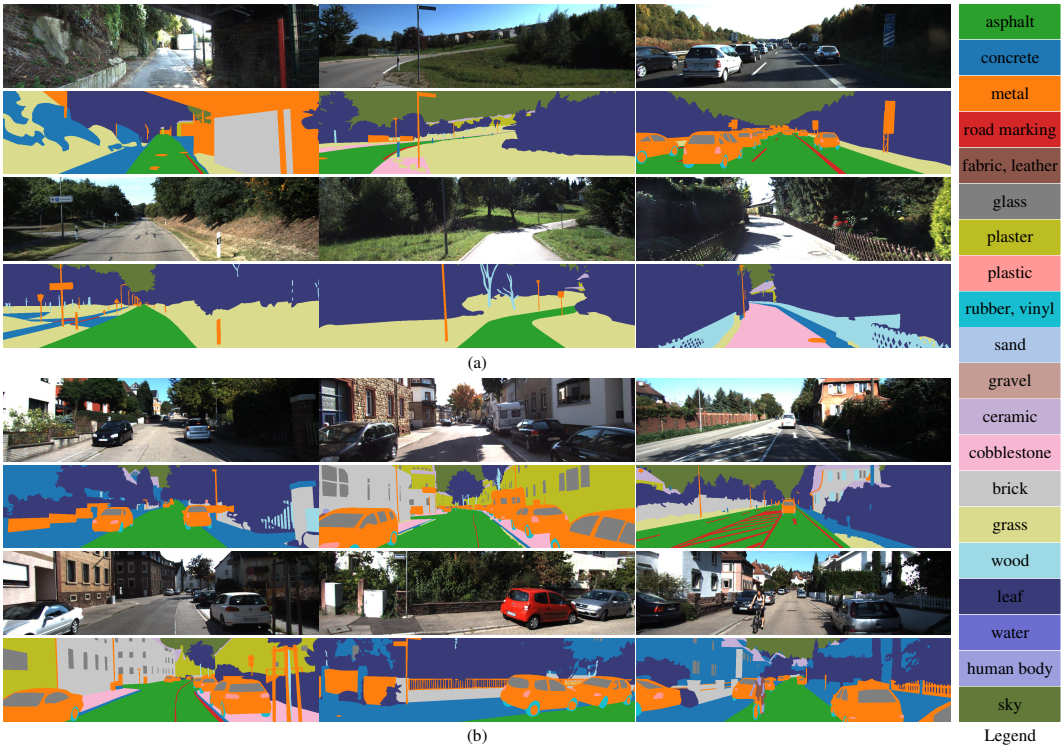


Figure 3.4: Visual examples of the test sets of (a) Split-1 and (b) Split-2. ©2022 Springer Nature [1]

imbalance in the material classes, which intensifies the difficulty for accurate RGB RMS. Figure 3.3 provides the overall pixel statistics *w.r.t.* each of the material categories, where 16 material categories span $0.9 \times 10^6 - 1.4 \times 10^8$ pixels, *i.e.*, 99.84% of the total number of pixels. In contrast, 4 categories including "sand," "gravel," "water," and "human body," accounts for 0.083%, 0.016%, 0.00026%, and 0.054% of the overall pixels, respectively. Table 3.1 reports the detailed per-class pixel statistics by scene IDs, where material categories "sand," "gravel," and "water" only show up in very few scenes, which is in line with the overall imbalanced pixel distribution.

For evaluation on KITTI-materials RGB RMS, an unexplored task, we define two training-test data splits, namely, *Split-1* and *Split-2*, where the test set of Split-1 (consists of scenes 0926019, 0926086, 0930034, and 1003047) contains more scenes with highways and rural areas while Split-2 (consists of scenes 0926064, 0926095, 0929004, and 0930016) is relatively biased to city scenes. Both splits use the all 1000 images of KITTI-Materials, where

800 images are for training and 200 images are for testing, but with different combinations of scenes. Figure 3.4 shows examples from the test sets of Split-1 and -2 and Table 3.1 reports the detailed per-class statistics of the Split-1 and -2, where both the splits preserve the diversity of materials in their training and test sets. Note that as some of the materials only appear in images of a few scenes, splits with all material categories in both training and test sets are difficult to realize when based on scenes except for the suggested Split-1 and -2.

3.4 RMSNet

We introduce RMSNet as a novel efficient yet effective baseline framework for RGB RMS. RMSNet extracts multi-level multi-scale features with hierarchical encoder(s) [4, 21, 22] equipped with the position-aware **F**eed-**F**orward **N**etwork (**FFN**) (three different encoder alternatives are evaluated in Section 3.5) and *effectively fuses texture and contextual cues of material appearance with the novel SAMixer model*. Figure 3.5 depicts the overview of RMSNet-MiT (*i.e.*, the default version of RMSNet which applies the Mix-Transformer-B2 [4] as the hierarchical encoder).

3.4.1 Hierarchical Feature Encoder

Hierarchical Feature Encoding. In this section, we employ Mix-Transformer-B2 (MiT-B2), *i.e.* the middle-size hierarchical Transformer encoder suggested by SegFormer [4] as the encoder to introduce the feature encoding process of RMSNet. The hierarchical encoder extracts a set of multi-level multi-scale feature maps, from 4 sequential learning hierarchies (*i.e.*, stages). Feature maps extracted from low to high hierarchy levels have high to low resolutions and contain gradually fewer local details of texture and more non-local context cues. For each hierarchy level, a layer that merges overlapping patches with the corresponding down-sampling ratio is employed to reduce the resolution of the input feature map. Given an input image of size $H_{in} \times W_{in} \times 3$, the encoder generates a set of hierarchical feature maps $\{\mathbf{X}_n\}$ with corresponding resolutions of $\{H_n \times W_n \times C_n\}$, where $n \in \{1, 2, 3, 4\}$ and C_n denotes the channel-size of $\mathbf{X}_n$. Note that we set $H_n \times W_n \times C_n = \frac{H_{in}}{2^{n+1}} \times \frac{W_{in}}{2^{n+1}} \times C_n$ by default.

Efficient MSA. To process high-resolution features efficiently, MiT-B2 employs *efficient MSA* which uses a **convolution** $\mathcal{F}_{R\times R}$ with kernel-size of $R \times R$ and stride of R to reduce the spatial resolutions of the key and value. For a given feature map $\mathbf{X} \in \mathbb{R}^{H\times W\times C}$ and its condensed feature map $\mathcal{F}_{R\times R}(\mathbf{X}) \in \mathbb{R}^{\frac{H}{R}\times\frac{W}{R}\times C}$, suppose that $\mathbf{Q}$, $\hat{\mathbf{K}}$, and $\hat{\mathbf{V}}$ denote the query, the key, and value transformed from $\mathbf{X}$ and $\mathcal{F}_{R\times R}(\mathbf{X})$, respectively. Efficient MSA (denoted by $\mathcal{A}(\cdot)$) becomes

$$\mathcal{A}\left(\mathbf{Q}, \hat{\mathbf{K}}, \hat{\mathbf{V}}\right) = \mathrm{softmax}\left(\frac{\mathbf{Q}\hat{\mathbf{K}}^T}{\sqrt{d}}\right)\hat{\mathbf{V}}\,. \tag{3.1}$$

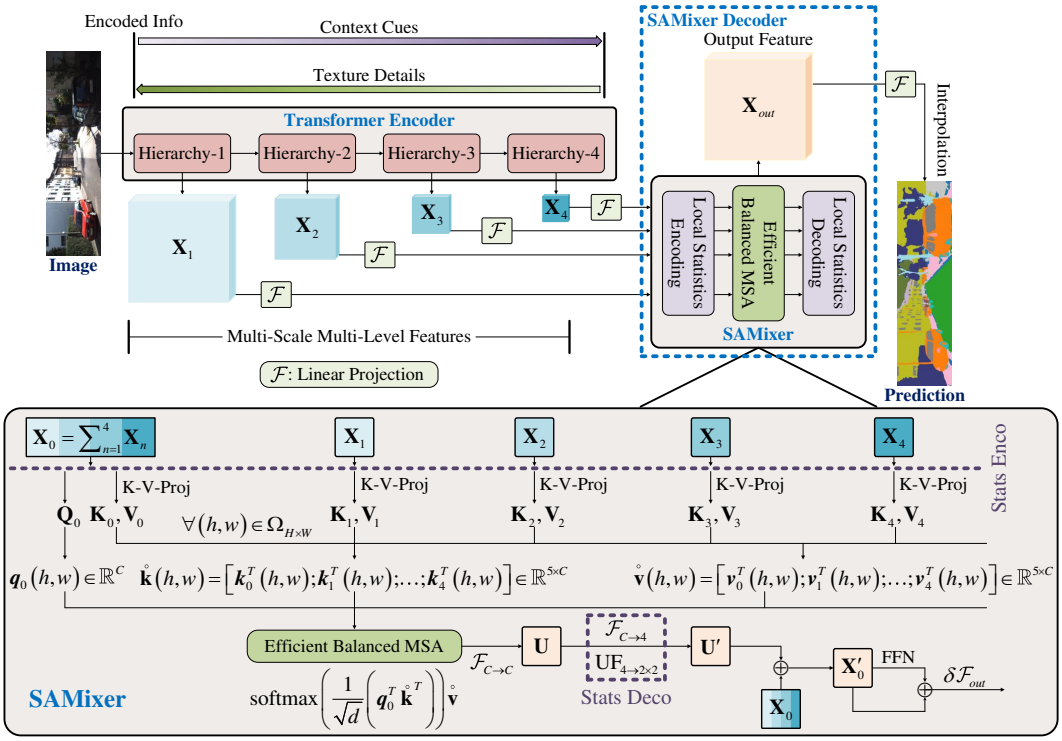


Figure 3.5: Overview of RMSNet(-MiT). "$\mathcal{F}$" denotes the linear (projection) layer with corresponding input and output channel sizes. "Q-Proj" and "K-V-Proj" are "Query-Projection" and "Key- and Value-projection," respectively. "Info" denotes "Information." We introduce our *Balanced MSA* operation in Sec. 3.4.2. "UF" means "Unfold" operation. "Stats," "Enco," and "Deco" denote local statistics, encoding, and decoding, respectively (*i.e.*, BLSED strategy in Sec. 3.4.2). After obtaining the output feature $\mathbf{X}_{out}$ of SAMixer, we employ a linear layer to generate the segmentation mask from $\mathbf{X}_{out}$ to achieve per-pixel material recognition. ©2022 Springer Nature [1]

Note that feature maps $\mathbf{X}$ and $\mathcal{F}_{R\times R}(\mathbf{X})$ are reshaped to the sizes $M \times C$ and $\frac{M}{R^2} \times C$ (*i.e.*, $M = HW$), respectively. Here, $d = \frac{C}{g}$, where g denotes the number of heads of MSA. The computational complexity of an MSA can be controlled with the resolution reduction ratio R. For hierarchy-1 to -4, MiT-B2 assigns $R = 8, 4, 2, 1$, respectively.

Position-aware FFN. MiT-B2 inserts a 3×3 **depth-wise convolution** $\mathcal{F}^{dw}_{3\times3}$ in each FFN, at the top of the first linear (projection) layer, to enforce position awareness without additional positional encodings. With this modification, local details can be preserved without sacrificing accuracy due to interpolation for matching resolutions. The position-aware FFN is defined as

$$\mathbf{X}'' = \mathcal{F}\left(\delta\left(\mathcal{F}^{dw}_{3\times3}\left(\mathcal{F}\left(\mathbf{X}'\right)\right)\right)\right) + \mathbf{X}', \tag{3.2}$$

where

$$\mathbf{X}' = \mathcal{F}\left(\mathcal{A}\left(\mathbf{Q}, \hat{\mathbf{K}}, \hat{\mathbf{V}}\right)\right) + \mathbf{X}, \tag{3.3}$$

denotes the attended feature merged with the shortcut residual (*i.e.*, $\mathbf{X}$) generated by the

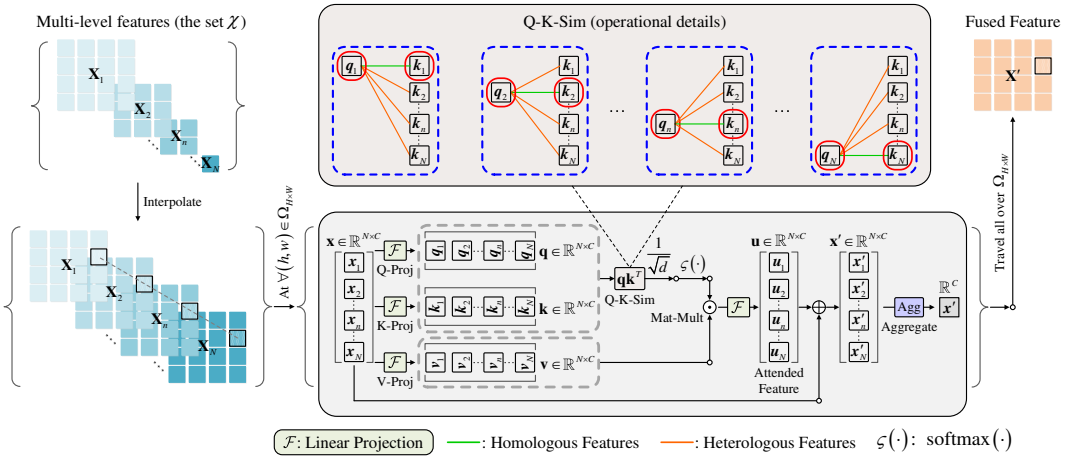


Figure 3.6: Self-attention-based (many-to-one) fusion for multi-level features. For a given pair of query and key feature vectors, "homologous" and "heterologous" features refer to the cases where the query and key are generated from the same and different source features, respectively. "Mat-Mult" denotes "Matrix Multiplication." "Agg" denotes the assigned aggregation scheme (*e.g.*, linear projection or weighted-summation) to merge all the attended feature vectors $\boldsymbol{x}'_n$ into a fused feature vector $\boldsymbol{x}'$ at each aligned position (h, w).

MSA layer and $\mathbf{X}''$ is the output feature of the FFN; δ denotes the assigned nonlinear activation (GELU [12] by default); $\mathcal{F}$ denotes the linear layer (*i.e.*, $\mathcal{F}_{1\times1}$).

3.4.2 SAMixer-based Decoder

Through careful examination of KITTI-Materials, we find that feature encoding and feature fusion are the two critical points for per-pixel material recognition (Section 3.5.3). Empirically, material annotations guide the powerful deep-learning encoders to extract discriminative texture features for different materials. The appearance of texture features, however, varies significantly with scale and occlusion. Structural dependencies and co-occurrences of local texture features may help extract a representation robust to this variability. Unlike semantic objects, however, materials often show more complicated spatial distributions (*i.e.*, more fragmented) and lack prominent shape cues. This makes the fusion of local textures and long-range context cues challenging. To realize effective fusion for Transformer-induced features, we propose a novel multi-level, multi-scale feature fusion model based on MSA, which we refer to as **SAMixer**. Figure 3.5 depicts the SAMixer-based decoder. SAMixer can efficiently fuse local and non-local features to generate robust representations for road scene materials.

Direct Feature Fusion with Self-Attention: A Discussion

Although the related problem of multi-scale feature learning with self-attention was discussed in recent works [37, 117], the problem of direct N-to-1 feature fusion through Q-K-V self-attention has been rarely explored. Here, we discuss how to realize effective feature

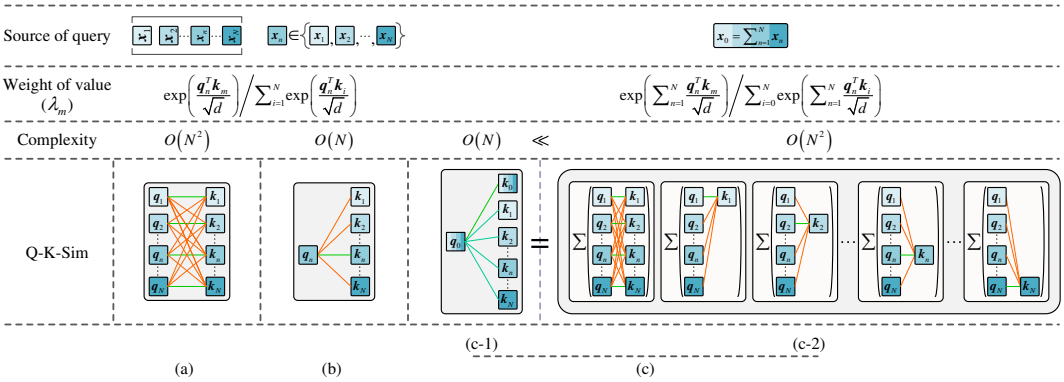


Figure 3.7: The three different types of Q-K-Sim measure for (many-to-one) MSA-based feature fusion: (a) the full Q-K-Sim measure of vanilla MSA; (b) the imbalanced partial Q-K-Sim measure triggered by a single query of $\boldsymbol{q}_n, n \in \{1, 2, \ldots, N\}$; (c-1) the proposed balanced, efficient Q-K-Sim measure of SAMixer, triggered by the query $\boldsymbol{q}_0$ from the container feature $\boldsymbol{x}_0$ constructed by summing each of the element features at the aligned position; (c-2) the expanded form of (c-1) with equivalent calculation results. In particular, by expanding a Q-K-V self-attention operation as a weighted-summation on the value vectors, we identify that (c-1)/(c-2) models richer query-key similarities in each individual weight computation than (a) and (b), and leaves room for the self-attention to learn more adaptive fusion of multi-level features. Further, (c-1) achieves the equivalent balanced Q-K-Sim of (c-2) with only $O(N)$ complexity, which is of only marginal additional computations to (b) and clearly more efficient than (a), by introducing the container query $\boldsymbol{q}_0$.

fusion with self-attention and investigate the main challenges of self-attention-based feature fusion. We then introduce our SAMixer in the subsequent subsections.

Excessive Computational Complexity. Self-attention introduces informative context dependencies to deep learning representations. For multi-scale feature fusion, however, it inevitably causes excessive computational overhead. Figure 3.6 illustrates the expected processing of a general self-attention-based feature fusion with operational details. Suppose that $\chi = \left\{ \mathbf{X}_n \in \mathbb{R}^{H_n \times W_n \times C_n} \mid n = 1, 2, \ldots, N \right\}$ is a set of feature maps for fusion ($N = 4$ in our experiments). The fused feature map $\mathbf{X}' \in \mathbb{R}^{H \times W \times C}$ is generated by mixing all element feature maps $\mathbf{X}_n \in \chi$ at each aligned position $(h, w) \in \Omega_{H \times W}$, where $\Omega_{H \times W}$ denotes the spatial lattice of $\mathbf{X}'$. Note that before fusion, each of the feature maps $\mathbf{X}_n$ of different sizes should be transformed and interpolated to the same size $H \times W \times C$ which we refer to as the *anchor size*.

With self-attention-based feature fusion, each element feature $\mathbf{X}_n$ is projected to $\mathbf{Q}_n$, $\mathbf{K}_n$, $\mathbf{V}_n$, where $\boldsymbol{q}_n\,(h, w)$, $\boldsymbol{k}_n\,(h, w)$, $\boldsymbol{v}_n\,(h, w)$ with the unified length C are corresponding feature vectors of $\mathbf{Q}_n$, $\mathbf{K}_n$, $\mathbf{V}_n$ at the given spatial position (w, h), respectively. We use $\boldsymbol{q}_n$, $\boldsymbol{k}_n$, $\boldsymbol{v}_n$ to denote $\boldsymbol{q}_n\,(h, w)$, $\boldsymbol{k}_n\,(h, w)$, $\boldsymbol{v}_n\,(h, w)$ if not specified. Similarly, We also omit the positional index (h, w) for the related features/descriptors (*e.g.*, $\boldsymbol{x}_n$). Without loss of generality, we discuss on the self-attention case of $g = 1$ (*i.e.*, single-head where $d = C$) for simplicity, which can be easily extended to the case of $g > 1$ (*i.e.*, multi-head). Then, to

fuse each feature vector at an aligned position (w, h), MSA can be defined as

$$\mathcal{A}(\mathbf{q}, \mathbf{k}, \mathbf{v}) = \text{softmax}\left(\frac{\mathbf{q}\mathbf{k}^T}{\sqrt{d}}\right)\mathbf{v}\,, \tag{3.4}$$

where $\mathbf{q} = [\boldsymbol{q}_n^T] \in \mathbb{R}^{N\times C}$ and $\mathbf{k}, \mathbf{v} = [\boldsymbol{k}_n^T], [\boldsymbol{v}_n^T] \in \mathbb{R}^{N\times C}$ are the query, key, and value at position (h, w), respectively, formed by concatenating the corresponding feature vectors along the row-axis in order.

Let $\mathbf{u} \in \mathbb{R}^{N\times C}$ denote the attended feature at position (h, w) computed by the MSA layer where $\boldsymbol{u}'_n \in \mathbb{R}^C$ is the n-th element feature vector of $\mathbf{u}$. Since $\mathbf{x}' = \mathbf{u} + \mathbf{x}$, the fused feature vector $\boldsymbol{x}' \in \mathbb{R}^C$ at (h, w) can be obtained by aggregating each of the element feature vectors $\boldsymbol{x}'_n \in \{\boldsymbol{x}'_n\}$ along the row-axis. In this manner, a general pure-MSA-induced feature fusion is estimated to have $O(N^2)$ computational complexity, which can be excessive since coarse features of local texture patterns are usually of large sizes. Moreover, raw MSA also necessitates an extra feature aggregation operation to help integrate the N attended features into a single representation.

Imbalanced Partial Q-K-Sim Measure. To reduce the complexity of MSA-based feature fusion, we propose to integrate all the N element features with only a single vector-matrix multiplication. The main challenge is that the full query-key similarity measure (*i.e.*, $\mathbf{q}\mathbf{k}^T$) uses different elements in the feature set χ in a balanced (*i.e.*, symmetric) manner (Figure 3.6) such that the informative cues in different source features are exploited comprehensively. In contrast, as illustrated in Figure 3.7, for each query vector $\boldsymbol{q}_n$, its corresponding decomposed (*i.e.*, partial) group of query-key similarity measure $\boldsymbol{q}_n^T\mathbf{k}^T$ is imbalanced for different keys. As a result, employing a decomposed group of Q-K-Sim measures independently, although efficient, leads to limited use of different source features and decreases the representation power. More specifically, by treating a self-attention as a weighted-summation about the value $\mathbf{v}$,

$$\begin{aligned}\text{softmax}\left(\frac{\boldsymbol{q}_n^T\mathbf{k}^T}{\sqrt{d}}\right)\mathbf{v} = \boldsymbol{\lambda}^T\mathbf{v} &= [\lambda_1, \lambda_2, \ldots, \lambda_N]\begin{bmatrix}\boldsymbol{v}_1^T\\ \boldsymbol{v}_2^T\\ \vdots\\ \boldsymbol{v}_N^T\end{bmatrix}\\ &= \sum_{m=1}^{N}\lambda_m\boldsymbol{v}_m^T\,,\end{aligned} \tag{3.5}$$

where the weight $\lambda_m \in \mathbb{R}$ assigned to the element value vector $\boldsymbol{v}_m$ of $\mathbf{v}$ is

$$\lambda_m = \frac{\exp\left(\frac{\boldsymbol{q}_n^T\boldsymbol{k}_m}{\sqrt{d}}\right)}{\sum_{i=1}^{N}\exp\left(\frac{\boldsymbol{q}_n^T\boldsymbol{k}_i}{\sqrt{d}}\right)}\,. \tag{3.6}$$

That is, in a partial MSA with respect to $\boldsymbol{q}_n$, the attention output can be biased by the information of $\boldsymbol{x}_n$ (*i.e.*, the corresponding source feature), as the $\boldsymbol{q}_n$ generated from $\boldsymbol{x}_n$ serves as the unique query to control the calculation of the weights. This induces an imbalanced feature fusion over different element features with self-attention resulting in insufficient exploitation of informative cues in the element features $\{\mathbf{X}_m\}$. To overcome this problem, we propose the efficient *Balanced Q-K-Sim Measure* which has a computational cost close to a partial Q-K-Sim Measure.

Proposed SAMixer

We construct **SAMixer** with MSA which has only an $O(N)$ computational complexity by deriving a new balanced query-key similarity (Q-K-Sim) measure. For this, a container feature is introduced by simply aggregating (*i.e.*, summing) all input features to trigger the lightweight MSA computation. SAMixer also introduces a new built-in bottleneck local encoding-decoding (BLSED) strategy to realize further efficiency and accuracy. Figure 3.5 depicts the diagram of SAMixer and Figure 3.7 illustrates the core idea of the balanced Q-K-Sim measure in SAMixer. In the following paragraphs, we present the two core components of SAMixer, *i.e.*, the balanced Q-K-Sim measure and BLSED strategy.

Balanced Q-K-Sim Measure. As discussed in Section 3.4.2, although efficient, the MSA induced by a partial Q-K-Sim measure causes imbalance leading to limited exploitation of texture and context cues within multi-level features. To address this problem, our goal is to calculate an efficient balanced MSA on the feature set χ with only one vector-matrix multiplication such that redundant computation is avoided while adequate adaptation is learned. We achieve this by introducing a novel query-key similarity measure which we refer to as the *balanced query-key similarity measure* (Figure 3.7(c)). The core idea of this balanced Q-K-Sim measure is the new tailored element feature referred to as the *container feature* that enables balanced computation on a single group of Q-K-Sim measure. We generate this container feature $\mathbf{X}_0 \in \mathbb{R}^{H\times W\times C}$ by aggregating each of the features in χ with a simple summation (*i.e.*, $\mathbf{X}_0 = \sum_{n=1}^{N} \mathbf{X}_n$). Then, the feature set χ can be expanded into a new set $\mathring{\chi}$ comprising of $N+1$ element features by including $\mathbf{X}_0$.

For $\forall\,(h,w) \in \Omega_{H\times W}$, we generate the key and value descriptors $\mathring{\mathbf{k}}, \mathring{\mathbf{v}} = [\mathring{\boldsymbol{k}}_n^T], [\mathring{\boldsymbol{v}}_n^T] \in \mathbb{R}^{(N+1)\times C}$ from element features in $\mathring{\chi}$ from features in $\mathring{\chi}$ and the single query vector $\boldsymbol{q}_0 \in \mathbb{R}^C$ from the container feature vector $\boldsymbol{x}_0 \in \mathbb{R}^C$, respectively. With this, we can compute an efficient balanced MSA on $\mathring{\chi}$ **(note that following we emphasize significant Equations by "dark blue")**

$$\mathcal{A}\left(\boldsymbol{q}_0, \mathring{\mathbf{k}}, \mathring{\mathbf{v}}\right) = \mathrm{softmax}\left(\frac{\boldsymbol{q}_0^T \mathring{\mathbf{k}}^T}{\sqrt{d}}\right)\mathring{\mathbf{v}}\,, \tag{3.7}$$

where the container-feature-induced Q-K-Sim measure $\boldsymbol{q}_0^T \mathring{\mathbf{k}}^T$ can be expanded, *i.e.*,

$$\begin{aligned}\boldsymbol{q}_0^T \mathring{\mathbf{k}}^T &= \left(\mathcal{F}^q\left(\boldsymbol{x}_0\right)\right)^T \mathring{\mathbf{k}}^T = \left(\mathcal{F}^q\left(\sum_{n=1}^{N} \boldsymbol{x}_n\right)\right)^T \mathring{\mathbf{k}}^T \\ &= \left(\sum_{n=1}^{N} \mathcal{F}^q\left(\boldsymbol{x}_n\right)\right)^T \mathring{\mathbf{k}}^T = \left(\sum_{n=1}^{N} \boldsymbol{q}_n\right)^T \mathring{\mathbf{k}}^T \\ &= \left(\sum_{n=1}^{N} \boldsymbol{q}_n^T\right) \mathring{\mathbf{k}}^T = \sum_{n=1}^{N} \boldsymbol{q}_n^T \mathring{\mathbf{k}}^T \,, \end{aligned} \tag{3.8}$$

by omitting the influence of the bias term in a linear projection (*i.e.*, suppose that the query projection $\mathcal{F}_q$ is a real linear transformation). Note that in Equation (3.8),

$$\mathcal{F}^q\left(\boldsymbol{x}_n\right) = \boldsymbol{q}_n, \forall n \in \{0, 1, \ldots, N\} \,, \tag{3.9}$$

as the query-projection layer is shared by all the element features in an MSA computation. As we can treat self-attention as a weighted-summation about the value $\mathring{\mathbf{v}}$ (*i.e.*, based on the Equations (3.5) and (3.6)) where the weight λ_m of $\boldsymbol{v}_m$ of $\mathring{\mathbf{v}}$, $\forall m \in \{0, 1, \ldots, N\}$ is

$$\begin{aligned}\lambda_m &= \frac{\exp\left(\frac{\boldsymbol{q}_0^T \boldsymbol{k}_m}{\sqrt{d}}\right)}{\sum_{i=0}^{N} \exp\left(\frac{\boldsymbol{q}_0^T \boldsymbol{k}_i}{\sqrt{d}}\right)} = \frac{\exp\left(\frac{\left(\sum_{n=1}^{N} \boldsymbol{q}_n^T\right) \boldsymbol{k}_m}{\sqrt{d}}\right)}{\sum_{i=0}^{N} \exp\left(\frac{\left(\sum_{n=1}^{N} \boldsymbol{q}_n^T\right) \boldsymbol{k}_i}{\sqrt{d}}\right)} \\ &= \frac{\exp\left(\sum_{n=1}^{N} \frac{\boldsymbol{q}_n^T \boldsymbol{k}_m}{\sqrt{d}}\right)}{\sum_{i=0}^{N} \exp\left(\sum_{n=1}^{N} \frac{\boldsymbol{q}_n^T \boldsymbol{k}_i}{\sqrt{d}}\right)} \,, \end{aligned} \tag{3.10}$$

unlike the Q-K-Sim measure of partial self-attention (*i.e.*, Equations (3.5) and (3.6)), our Q-K-Sim measure induces a balanced Q-K-V self-attention with only a single vector-matrix multiplication, $\forall m$. With this, our MSA-based feature fusion makes effective use of the scale-aware multi-level texture and context cues within varied element features while reducing the quadratic complexity of $O(N^2)$ to only $O(N)$. Figure 3.7(a), (b), and (c) illustrate the *full Q-K-Sim of the raw MSA*, the *imbalanced decomposed Q-K-Sim*, and the *efficient balanced Q-K-Sim in our SAMixer*, respectively. Note that although the simpler Q-K-Sim measure $\boldsymbol{q}_0^T \mathbf{k}^T$ (*i.e.*, exclude the container key vector $\boldsymbol{k}_0$ in $\mathring{\mathbf{k}}$) is also balanced with different element features, we include $\boldsymbol{k}_0$ to increase the adaptability of the self-attention.

It is worth noting that the balanced Q-K-Sim measure not only reduces the complexity of MSA, but also models richer Q-K interactions than the raw MSA mechanism in

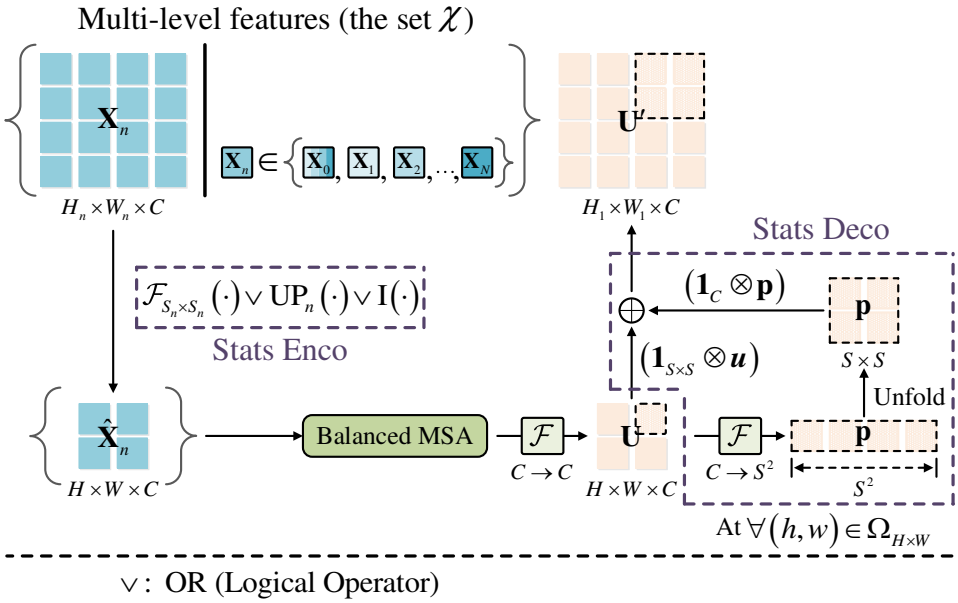


Figure 3.8: Operational diagram of the BLSED strategy. "Stats," "Enco," and "Deco" denote local statistics, encoding, and decoding, respectively. To encode the condensed feature map of local statistics, we employ corresponding $2D$ convolutional layer $\{\mathcal{F}_{S_n \times S_n}\}$ with kernel size and stride of $S_n \times S_n$, bilinear interpolation UP_n, or identical mapping I to each feature map with higher, lower, or identical resolution to the given anchor size $H \times W \times C$. After the fusion of the condensed multi-level feature maps with the balanced MSA computation, we decode the condensed fused feature map $\mathbf{U}$ to the high-resolution fused feature map $\mathbf{U}'$ with incorporating additional local cues.

each individual similarity computation (Figure 3.7). As a result, our SAMixer model effectively fuses multi-resolution multi-level features to produce discriminative representations for road scene materials.

BLSED Strategy. We achieve further efficiency and accuracy by introducing a lightweight embedded encoder-decoder strategy in SAMixer. Figure 3.8 depicts the process of BLSED strategy. We first assign an anchor size $H \times W \times C$, where $H = \frac{H_1}{2^l}$ and $W = \frac{W_1}{2^l}$. Here, $l \in \mathbb{Z}^+$; H_1 and W_1 are the largest height and width of all the input features extracted by the hierarchical encoder. Note that we apply $l = 1$ such that $S = 2^l = 2$ by default in our experiments.

Before MSA, we encode local statistics $\hat{\mathbf{X}}_n \in \mathbb{R}^{H \times W \times C}$ from each of the input feature maps $\mathbf{X}_n$ in $\mathring{\chi}$ whose spatial resolutions $H_n \times W_n$ are higher than $H \times W$ by employing corresponding $2D$ convolutions $\{\mathcal{F}_{S_n \times S_n}\}$ with strides of $\{S_n\}$ ($S_n = \frac{H_n}{H}$ is divisible by 2). To reduce computational cost, each $\mathcal{F}_{S_n \times S_n}$ is replaced by splicing a depth-wise convolution with a linear layer (*i.e.*, $\mathcal{F}\left(\mathcal{F}^{dw}_{S_n \times S_n}\right)$). We interpolate all the feature maps to the anchor size whose spatial resolution is smaller than $H \times W$. We preserve the size of any feature naturally possessing spatial resolution of $H \times W$. Since the anchor size is smaller than the highest resolution of features, we produce the container feature by $\mathbf{X}_0 = \sum_{n=1}^{N} \mathrm{UP}_n(\mathbf{X}_n)$, where UP_n denotes up-sampling by bilinear interpolation with a scale

factor of $\frac{H_1}{H_n}$ (equivalent to $\frac{W_1}{W_n}$ in our experiments). For $n = 1$, UP degrades to an identity mapping.

After MSA, we decode the attended fused feature map $\mathbf{U} \in \mathbb{R}^{H\times W\times C}$ to a high-resolution feature map $\mathbf{U}' \in \mathbb{R}^{H_1\times W_1\times C}$ with a channel-spatial decoupled combination scheme which significantly reduces the computational cost. We achieve this by decoding each attended feature vector $\boldsymbol{u} \in \mathbb{R}^C$ at position $(h, w) \in \Omega_{H\times W}$ to produce the local descriptor $\mathbf{p} \in \mathbb{R}^{S^2}$, with a linear projection $\mathcal{F}$ with input channels of C and output channels of S^2. Then, each local descriptor is unfolded to form the local feature patch of size $S \times S$ and we mix each local patch with its corresponding attended feature vector $\boldsymbol{u}$ to produce $\boldsymbol{u}'$ (*i.e.*, the feature vector on $\mathbf{U}'$ at (h, w)):

$$\mathbf{u}' = (\mathbf{1}_{S\times S} \otimes \boldsymbol{u}) \oplus (\mathbf{1}_C \otimes \mathbf{p}) \,, \tag{3.11}$$

where $\otimes$ denotes Kronecker product; $\mathbf{1}_{S\times S}$ and $\mathbf{1}_C$ each denotes a ones matrix of the corresponding size. We then obtain $\mathbf{U}'$ by arranging each of the mixed local feature patches $\mathbf{u}'$ according to their spatial position order.

As each attended feature vector $\boldsymbol{u}$ is generated from the fused local statistics with regional context and the feature patch $\mathbf{p} \in \mathbb{R}^{S\times S}$ leaves enough room to learn the local details to compensate the attended feature vector, the decoded high-resolution feature map $\mathbf{U}'$ contains rich mixtures of local and long-range cues for accurate RMS. With the BLSED strategy, SAMixer also achieves higher efficiency by operating MSA on the condensed feature maps of local statistics.

Segmentation Mask Generation. The output feature $\mathbf{X}_{out} \in \mathbb{R}^{H_1\times W_1\times C}$ of SAMixer is generated by applying a linear layer $\mathcal{F}_{out}$ (with a GELU activation δ) over the output of the FFN

$$\mathbf{X}_{out} = \delta\left(\mathcal{F}_{out}\left(\mathcal{F}\left(\delta\left(\mathcal{F}\left(\mathcal{F}^{dw}_{3\times 3}\left(\mathbf{X}'_0\right)\right)\right)\right) + \mathbf{X}'_0\right)\right) \,, \tag{3.12}$$

where

$$\mathbf{X}'_0 = \mathbf{U}' + \mathbf{X}_0 \,. \tag{3.13}$$

As for the FFN, unlike MiT-B2, in SAMixer we plug the depth-wise convolution before the first linear layer ($\mathcal{F}$) to learn positional cues, which increases the inference speed of the FFN. The segmentation mask is obtained by employing a linear layer (*i.e.*, $\mathcal{F}_{out}$) with an output channel-size of the number of material classes (*i.e.*, 20) on $\mathbf{X}_{out}$.

3.5 Experiments & Discussions

We evaluate the effectiveness of our RMSNet(s) on the KITTI-Materials dataset with detailed ablation studies and thorough comparisons with past material segmentation methods [79, 85], road scene semantic segmentation methods with CNN encoders [98, 3, 99, 96, 11, 114], related state-of-the-art Transformers/ConvNeXt [19, 20, 4, 21, 22] and a gating-based dynamic network [17] that has been applied to semantic segmentation. Note that

DeepLabv3+ [3] also represents MCubeSNet [106] without semantic segmentation masks and using RGB only.

3.5.1 Implementation Details

Two different training-test data splits (denoted by *Split*-1 and -2, respectively) of KITTI-Materials with different characteristics are used for evaluation. The test set of "split-1" contains more scenes with highways and rural areas while "split-2" is biased to city scenes (as detailed in Sec. 3.3). Both splits consist of all 1000 images of KITTI-Materials where 800 images are used for training and 200 images are preserved for testing with different split rules. For all models, we apply the AdamW optimizer with a weight decay of 0.01 for 300 epochs including 10 epochs of linear warm-up. Following [4], we start the learning rate from 6×10^{-5} and 6×10^{-4} for encoders and decoders, respectively, with a cosine decay scheduler and a mini-batch of 16. We adopt standard image augmentation settings [3]. In the training phase, images are randomly center-cropped and then resized to 512×512 pixels, while in the testing, images are fixed to the original size (*i.e.*, 320×1216 pixels). To reduce the negative effect of extreme data imbalance, we calculated balancing weights based on class frequencies of materials and applied them to CE-losses of all models. Experiments are conducted on a computer with $4 \times$ RTX A5000 GPUs. For fair comparisons, all encoders of our method and compared methods use ImageNet [119] pre-trained weights obtained from corresponding open-sourced projects or websites. All methods are evaluated in the raw image size without multi-scale averaging augmentation [120]. We use the mean intersection of union (mIoU) to evaluate the performance of each model.

3.5.2 Main Results

Based on the benchmark KITTI-Materials dataset, we verify the effectiveness of our network designs by comparing with (1) existing general material segmentation methods for RGB images [79, 85]; (2) popular road scene semantic segmentation methods with CNN encoders [98, 3, 99, 96, 11]; (3) enhanced DeepLabv3+ [3] with a multi-scale fusion method (*i.e.*, SKNet [114]) and a SOTA gating-induced dynamic networks [17]; (4) related SOTA networks (*i.e.*, Transformers [21, 19, 20, 4] and ConvNeXt [22]) that have been validated on semantic segmentation/recognition. Note that we validate our RMSNet/SAMixer module with three different RMSNet variants that apply the SOTA network backbones MixTransformer-B2 [4], DaViT-T [21], and ConvNeXt-T [22] (namely, RMSNet-MiT, RMSNet-DaViT, and RMSNet-ConvNeXt), respectively.

Table 3.2 reports the results of experimental comparison of our RMSNet(s) and baseline methods. Main conclusions drawn from these results include that (1) all the three different RMSNet variants enjoy clear improvements over all the other methods for general material segmentation and road scene semantic segmentation in accuracy; RMSNet variants equipped with the novel SAMixer module demonstrate significant improvements

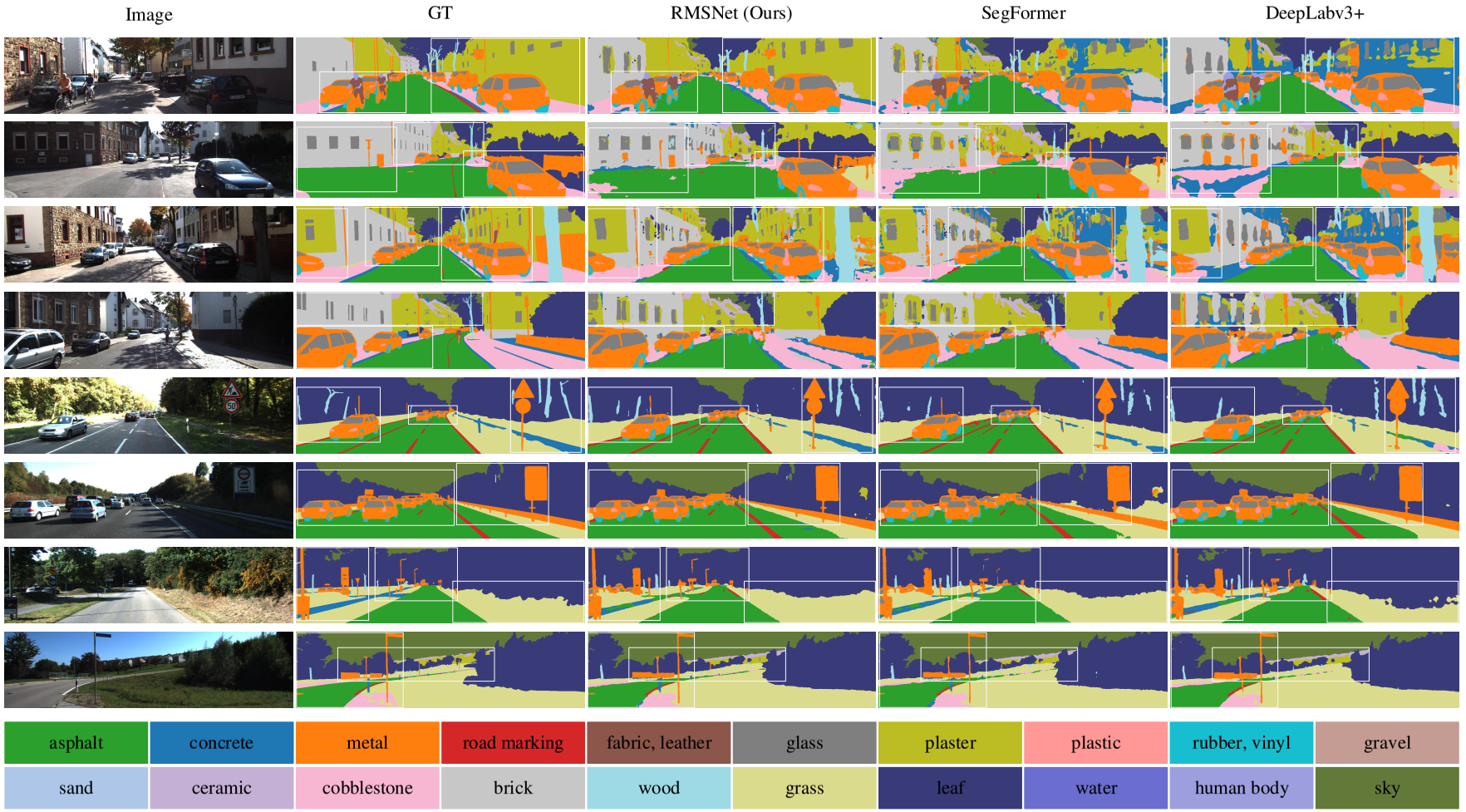


Figure 3.9: Examples of visualized segmentation results on KITTI-Materials, compared with DeepLabv3+ [3] and SegFormer [4]. "GT" denotes "Ground Truth." Our RMSNet(-MiT) produces cleaner segmentations on various materials critical in road scene understanding. ©2022 Springer Nature [1]

Table 3.2: RGB road scene material segmentation results on KITTI-Materials dataset for different methods. (1) "Trs," "Lt," and "F" denote "Transformer," "Light," and "Full," respectively. (2) "RMSN," "DVT," and "CNXt" denote "RMSNet," "DaViT," and "ConvNeXt," respectively. (3) "SEG" denotes "Segmentation." Methods with the suffix "-SEG" denote the segmentation networks built with the corresponding SOTA (MetaFormer) backbones (*i.e.*, ViT [19], CvT [20], DaViT [21], and ConvNeXt [22]) with the SOTA All-MLP decoder [4] that applies all the encoded levels of feature. "-D" denotes "-Decoder." ‡: Methods whose original code cannot support multi-GPU training/inference settings. ©2022 Springer Nature [1]

Method	Encoder	SEGHead	#Params	FPS	mIoU(%)↑	
					Split-1	Split-2
MINC‡ [79]	VGG16 [81]	MINC [79]	134.34M	15.88	29.73	32.12
MCT‡ [85]	VGG16 [81]	MCT [85]	25.42M	7.40	30.87	33.16
PSPNet (Lt) [96]	ResNet101 [11]	PSP (Lt) [96]	43.38M	14.22	31.92	37.11
PSPNet (F) [96]	ResNet101 [11]	PSP [96]	65.58M	13.31	42.04	46.62
DeepLabv3 [98]	ResNet101 [11]	ASPP [98]	58.04M	14.49	39.40	43.41
DeepLabv3+ [3]	ResNet101 [11]	ASPP-D [3]	59.34M	14.60	41.35	46.09
	SK-ResNet101 [114]	ASPP-D [3]	60.47M	14.08	41.96	46.04
DeeperLab [99]	ResNet101 [11]	DeeperLab-D [99]	240.58M	11.29	42.56	47.12
DDF-DL [17]	DDFNet101 [17]	ASPP-D [3]	42.94M	12.66	41.55	46.41
ViT-SEG [19]	ViT-B/16 [19]	ALL-MLP [4]	89.03M	13.69	40.02	46.06
CvT-SEG [20]	CvT-13 [20]	ALL-MLP [4]	21.89M	18.02	41.72	47.54
SegFormer [4]	Mix-Trs-B2 [4]	ALL-MLP [4]	27.36M	18.87	44.47	48.32
RMSN-MiT (Ours)		**SAMixer (Ours)**	31.53M	16.81	**46.82**	**50.34**
DaViT-SEG [21]	DaViT-T [21]	ALL-MLP [4]	31.08M	18.59	43.86	48.25
RMSN-DVT (Ours)		**SAMixer (Ours)**	35.25M	17.58	**45.84**	**49.76**
ConvNeXt-SEG [22]	ConvNeXt-T [22]	ALL-MLP [4]	31.31M	19.06	45.58	50.04
RMSN-CNXt (Ours)		**SAMixer (Ours)**	35.48M	17.43	**47.25**	**51.30**

to the corresponding segmentation networks that apply the SOTA **M**ulti-**Le**v**e**l (MLv) All-MLP decoder [4] using identical backbones, yet with similar efficiency. These demonstrate the effectiveness of our network designs for RGB RMS, which leverages effective hierarchical feature encoding and adaptive feature fusion to generate robust representations for various road scene materials. It is worth noting that the SOTA MetaFormer derivatives ConvNeXt [22] and DaViT [21] are both demonstrated to introduce clear accuracy gains to Mix-Transformer [4] for the recognition/segmentation of semantic objects. They, however, show slight improvements or are even inferior in accuracy to Mix-Transformer (SegFormer) on KITTI-Materials RGB RMS. This demonstrates the significance of effective multi-level feature fusion for realizing accurate RGB RMS. Further analysis of the critical clues and evidence for our designs are discussed in Section 3.5.3. ***Note that by taking into account***

Table 3.3: Per-class comparative results of our models and other methods on KITTI-Materials. (1) "RMSN," "DVT," and "CNXt" denote "RMSNet," "DaViT," and "ConvNeXt," respectively. (2) "road mk," "fab, lthr," "rubr, vl," "cob," and "hum bd" denote "road marking," "fabric, leather," "rubber, vinyl," "cobblestone," and "human body," respectively.

Method	Split	asphalt	concrete	metal	road mk	fab, lthr	glass	plaster	plastic	rubr, vl	sand	gravel	ceramic	cob	brick	grass	wood	leaf	water	hum bd	sky	mean
DeepLabv3	1	77.47	27.41	53.73	43.84	35.81	48.23	42.53	31.62	29.84	0	0	40.22	50.24	22.58	68.32	30.13	83.92	0	14.66	87.47	39.40
DeepLabv3+		79.58	29.29	56.74	53.74	34.03	50.55	44.07	30.88	40.51	0	0	41.60	40.53	26.55	71.50	30.29	**85.92**	0	20.24	91.03	41.35
DeeperLab		81.13	26.45	55.79	58.80	43.72	56.52	54.95	33.36	43.73	0	0	40.96	40.97	21.10	71.53	27.85	85.34	0	18.47	90.32	42.56
DDF-DL		77.44	26.48	55.02	51.21	36.86	54.98	45.29	40.91	41.00	0	0	41.31	41.63	19.03	**72.24**	25.38	85.57	0	24.05	92.64	41.55
ViT-SEG		81.67	24.12	52.88	51.17	30.11	53.68	45.43	38.42	35.76	0	0	42.30	37.92	13.36	69.42	25.23	83.47	0	22.84	92.64	40.02
CvT-SEG		83.38	26.75	55.04	51.35	37.66	54.34	38.62	36.65	41.28	0	0	43.57	48.15	25.46	62.14	30.10	82.40	0	24.35	93.23	41.72
SegFormer		82.67	28.47	57.81	58.59	36.46	60.54	48.36	**43.83**	47.09	0	0	**48.28**	51.85	25.54	65.91	31.32	83.70	0	24.38	94.54	44.47
RMSN-MiT		**85.14**	**29.58**	**58.66**	**60.65**	**46.69**	**60.75**	**56.12**	42.91	**48.79**	0	0	45.47	**57.62**	**31.25**	69.62	**35.47**	85.31	0	**27.55**	**94.89**	**46.82**
DaViT-SEG		82.32	28.46	58.85	**58.43**	38.81	51.53	40.12	42.21	44.48	0	0	**45.86**	53.84	**25.58**	68.96	**32.80**	84.62	0	26.05	94.29	43.86
RMSN-DVT		**84.77**	**32.42**	**60.06**	57.63	**42.66**	**55.82**	**56.25**	**46.20**	**47.67**	0	0	44.74	**55.18**	24.95	**71.91**	26.40	**86.52**	0	**29.09**	**94.52**	**45.84**
CNXt-SEG		86.04	29.27	**61.07**	**63.84**	42.50	58.82	45.72	41.35	48.12	0	0	47.04	58.60	23.88	74.16	23.73	**87.29**	0	26.22	93.94	45.58
RMSN-CNXt		**86.16**	**30.47**	58.40	61.22	**42.79**	**60.08**	**48.11**	**48.07**	**49.36**	**12.50**	0	**51.72**	**62.06**	**26.57**	**74.27**	**24.37**	87.00	0	**27.81**	**94.12**	**47.25**
DeepLabv3	2	81.66	18.47	55.84	38.39	50.87	56.53	39.69	34.98	35.11	0	0	50.55	36.61	46.32	76.10	32.57	88.85	0	37.69	87.92	43.41
DeepLabv3+		85.66	15.79	60.24	54.33	48.16	62.82	41.95	40.35	41.06	**0.28**	0	53.01	33.63	**50.12**	77.82	35.21	90.42	0	38.32	92.62	46.09
DeeperLab		86.56	18.65	61.10	54.43	51.30	65.41	41.99	42.40	43.81	0	0	53.89	32.80	49.14	77.92	37.26	90.25	0	43.31	92.38	47.12
DDF-DL		**87.63**	18.43	60.91	56.22	50.99	61.32	43.10	43.46	43.20	0	0	54.55	30.94	43.43	77.48	36.14	90.04	0	38.00	92.31	46.41
ViT-SEG		83.58	15.93	56.67	**59.45**	48.68	61.47	46.66	40.42	38.13	0	0	53.53	38.08	46.94	74.41	30.86	90.28	0	42.98	93.19	46.06
CvT-SEG		82.92	21.28	58.95	53.79	**57.20**	61.25	41.16	40.97	40.89	0	0	54.24	**52.39**	50.01	78.63	31.43	90.54	0	47.28	87.82	47.54
SegFormer		85.08	**22.87**	60.43	56.99	55.61	64.86	38.24	42.48	44.72	0	0	54.24	52.38	40.60	76.48	38.30	91.03	0	48.22	**93.80**	48.32
RMSN-MiT		86.51	22.84	**61.81**	58.51	52.56	**67.17**	**48.12**	**48.50**	**47.87**	0	0	**60.80**	51.05	47.72	**79.19**	**40.77**	**91.31**	0	**49.10**	92.93	**50.34**
DaViT-SEG		84.15	19.41	60.12	56.95	56.65	63.90	44.79	40.99	46.24	0	0	55.93	**46.72**	**47.55**	75.79	37.94	90.73	0	**43.93**	**93.17**	48.25
RMSN-DVT		**85.92**	**22.29**	**62.52**	**58.59**	**59.77**	**65.93**	**48.73**	**49.10**	**48.05**	0	0	**58.10**	45.89	46.71	**77.16**	**39.81**	**91.20**	0	42.94	92.41	**49.76**
CNXt-SEG		86.85	16.27	63.64	**60.55**	61.25	65.71	42.27	**48.70**	49.08	0	0	59.68	36.01	54.17	77.89	40.92	91.54	0	52.69	**93.66**	50.04
RMSN-CNXt		**87.30**	**19.56**	**65.27**	59.98	**63.29**	**67.47**	**43.82**	48.25	**51.17**	0	0	**59.82**	**40.16**	**57.07**	**78.84**	**44.11**	**91.83**	0	**54.65**	93.48	**51.30**

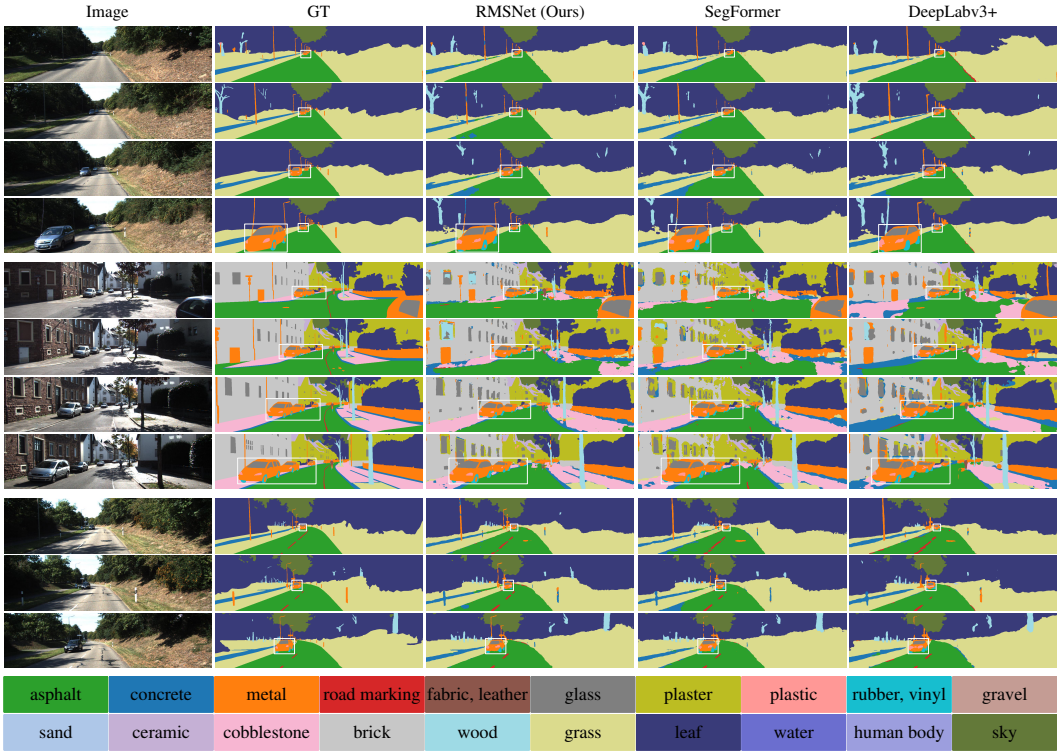


Figure 3.10: Example segmentation results for moving cars at different scales. Three different groups of examples are provided. Our RMSNet(-MiT) produces richer details of the contours and shapes of objects composed of multiple materials. ©2022 Springer Nature [1]

the accuracy-complexity trade-off, we select RMSNet-MiT as the representative (i.e., default version) of the RMSNet variants and denote it as RMSNet in the subsequent text if not specified.

To demonstrate detailed performance differences on each material, we report the per-class results of experimental comparisons in Table 3.3 of our RMSNet(-MiT) and other popular/SOTA methods that show competitive accuracy. Results on Split-1 show that our RMSNet yields the best IoU on most material classes including "asphalt," "concrete," "metal," "road marking," "fabric, leather," "glass," "plaster," "rubber, vinyl," "cobblestone," "brick," "wood," "human body," and "sky," where RMSNet introduces clear gains on "asphalt," "road marking," "rubber, vinyl," "plaster," "metal," "cobblestone," "brick," "wood," and "human body," which are materials critical in road scene understanding. Consistent with the results on Split-1, our RMSNet shows the highest IoU on most material classes, where it achieves clear improvements on "metal," "glass," "plaster," "plastic," "rubber, vinyl," "ceramic," "grass," "wood," and "human body," which are common material categories in city road scenes. Enhanced by the proposed SAMixer model, our RMSNet variants RMSNet-DaViT and -ConvNeXt both enjoy significant improvements in IoU to DaViT-SEG and ConvNeXt-SEG with the SOTA All-MLP decoder on most material classes.

Note that all of the compared methods fail to correctly predict materials "sand," "gravel,"

and "water," since these three materials have no salient appearance characteristics, and are extremely rare in both the training and test sets. For example, the material "water" appears in only around 1×10^3 pixels which account for approximately 0.00026% of the overall pixels. Additionally, the material "brick" is also challenging for all methods, since there are very few pixels and its appearance is close to other materials in road scenes that contain far more pixels (*e.g.*, "asphalt" and "concrete"). Results on materials "leaf," "grass," and "sky" also show very low variances, since these materials have comparatively subtle textures and occupy a large number of pixels in the training and test sets. For example, the material "leaf" contains about 1.33×10^8 pixels, which accounts for more than 34% of all pixels, and is over 1.3×10^5 times more than the number of pixels of the material "water."

We show visual examples in Figure 3.9, where we find that our RMSNet(-MiT) outperforms competing baseline DeepLabv3+ [3] and the SOTA SegFormer [4] by a clear margin on categories "fabric," "glass," "metal," "rubber," and "human body." These materials span a wide range of appearances as part of different semantic objects (*e.g.*, vehicles, bicycles, road markings, and pedestrians). Figure 3.10 demonstrates the significance of incorporating tailored texture-context feature fusion for road scene material segmentation through visual comparison between RMSNet and the compared methods. Our RMSNet with the SAMixer module achieves cleaner segmentation on windows, headlights, vehicle bodies, and wheels of moving cars of different scales (*i.e.*, different distances from viewpoint).

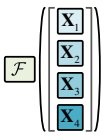
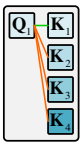
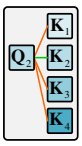
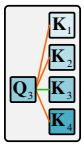
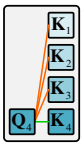
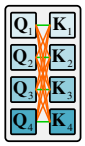
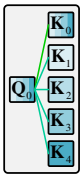
3.5.3 Ablation Study

Using split-1 of the KITTI-Materials dataset, we conduct targeted ablation studies on the proposed SAMixer and its core ingredients, *i.e.*, the *balanced query-key similarity measure* and *bottleneck local encoding-decoding strategy*, to verify their effectiveness in increasing efficiency and accuracy.

Balanced Q-K-Sim Measure

Here we analyze and validate the proposed balanced Q-K-Sim measure. We conduct experimental comparison by introducing three targeted control groups of feature fusion models built on the vanilla MSA mechanism with the raw full Q-K-Sim measure (denoted by "SAM-MSA-Full"); a series of imbalanced partial MSA mechanisms where the queries are only generated from one of the element feature maps of 4 different levels for fusion (denoted by "SAM-Imb-1" to "Imb-4"); and the SOTA All-MLP decoder of SegFormer [4] which leverages multi-level features through the linear aggregation. Note that **"SAMixer" denotes our original SAMixer model with the proposed balanced Q-K-Sim measure.** All the compared feature fusion models are constructed with the BLSED strategy to avoid the explosion in memory consumption with MSA computation, and we show results of SAM-MSA-Full with two different aggregation schemes (*i.e.*, the raw linear projection and

Table 3.4: Ablation studies on the balanced Q-K-Sim measure. (1) "SAM" and "Imb" denote "SAMixer" and "Imbalanced," respectively. (2) Note: "$\mathcal{F}$" and "Gating" denote the feature aggregations through *Linear Projection* and *self-gating*, respectively. As the main feature of the method *SAM-MSA-Full* is of size $\mathbb{R}^{4\times C\times H\times W}$ while the required fused feature size is $\mathbb{R}^{C\times H\times W}$, it requires an extra aggregation strategy to merge each of the attended feature elements such that the dimension-1 is squeezed from 4 to 1.

Method	SegFormer	SAM-Imb-1	SAM-Imb-2	SAM-Imb-3	SAM-Imb-4	SAM-MSA-Full		**SAMixer**
								
						LP	Gating	
Encoder	Mix-Transformer-B2 [4]							
Complexity	$O(N)$	$O(N)$				$O(N^2)$		$O(N)$
#Parameters	27.36M	31.53M	31.53M	31.53M	31.53M	33.94M	33.94M	31.53M
FPS	18.87	17.51	17.40	17.25	17.32	12.56	12.30	16.81
mIoU(%)↑	44.47	44.89	44.73	45.48	45.34	**45.15**	**46.41**	**46.82**

the weighted-summation with softmax-based self-gating suggested by [121]). Table 3.4 shows the results which show three characteristics.

1. Our original SAMixer enjoys significant improvements in accuracy over the modified SAMixers with imbalanced partial MSAs (*i.e.*, SAM-Imb-1 to Imb-4), which clearly demonstrates the effectiveness of the proposed balanced Q-K-Sim measure.
2. Our original SAMixer shows superior accuracy to the counterparts with raw full Q-K-Sim measures using aggregation schemes of direct linear projection (*i.e.*, "LP") and self-gating-based weighted-summation (*i.e.*, "Gating"). In particular, the original SAMixer achieves this with far fewer computational costs.
3. SAM-Imb-1 and -2 show close results to the raw SegFormer while SAM-Imb-3 and -4 still improve SegFormer. These results demonstrate that the features extracted from the deeper levels are relatively more informative to work as the main feature for fusion, and the feature fusion framework of SAMixer (*i.e.*, MSA mechanism with embedded local bottleneck statistics encoding-decoding) can be more effective in exploiting informative multi-level features than the direct linear projection (*i.e.*, the core of the All-MLP decoder).

The comparisons among the original SAMixer and different control groups demonstrate that the comprehensive modeling of query-key relationships in each individual weight computation for the value (described in Sec. 3.4.2 and 3.4.2) is non-trivial to MSA-based multi-level feature fusion. SAM-MSA-Fulls finalizes the direct fusion of multi-level features

outside the operation of Q-K-V self-attention with an extra aggregation scheme (*i.e.*, "LP" or "Gating"). These aggregation schemes, however, may not ensure full exploitation of the informative cues of attended Transformer-Induced Features (**Ti-Feats**). For instance, "LP" is expected to learn adaptive fusion with extensive parameters, but shows marginal accuracy gains to the raw SegFormer, *i.e.*, it compromises the MSA mechanism of SAMixer. In contrast, the original SAMixer introduces significant gains in accuracy to the SOTA All-MLP decoder which mixes multi-level features through linear aggregation. This clearly shows that effective joint modeling of fusion within Q-K-V self-attention can make more comprehensive use of multi-level Ti-Feats than the cooperation of a series of Q-K-V self-attention branches with an outside linear projection. This confirms the effectiveness of our proposed multi-scale feature fusion model.

BLSED Strategy

Effectiveness. We evaluate the effectiveness of our BLSED strategy by comparing the original SAMixer (denoted by "SAMixer") with two targeted control groups the abridged SAMixer(s) without the BLSED strategy (denoted by "SAMixer-a"), and the All-MLP decoder of SegFormer. Note that for SAMixer-a, to prevent excessive computational cost, we follow the resolution reduction strategy in Mix-Transformer layer to apply a lightweight depth-wise convolutional layer (*i.e.*, *DW*), and a heavyweight vanilla convolutional layer (*i.e.*, *V*), respectively. The results are reported in Table 3.5. Our original SAMixer outperforms the abridged SAMixers in both accuracy and efficiency. It also improves the All-MLP decoder of the SOTA SegFormer by a clear margin on accuracy. This verifies the effectiveness of our BLSED strategy.

Reduction Ratio Setting. We assign a unified resolution reduction ratio (denoted by "Ratio") as the stride and kernel size for corresponding depth-wise convolutions to control the encoding process of the local statistics of feature maps for fusion. We conduct this ablation study to evaluate the effectiveness of the BLSED strategy with different Ratio settings. Table 3.6 reports the comparative results with reduction ratios of "W/o (DW)", 2, and 4, where "W/o (DW)" denotes removing BLSED strategy and applying a depth-wise convolution to perform resolution reduction instead. Based on this evaluation, we set "ratio=2" by default, since it reaches high accuracy with competitive efficiency, compared with other settings.

Collaboration of Effective Hierarchical Feature Encoding and Adaptive Fusion

We discuss the two particular experimental results that provide critical evidence of the effectiveness of our major idea of collaborative application of faithful multi-level feature extraction and adaptive feature fusion.

Multi-level feature matters significantly. Table 3.7 reports the comparative results of a series of **abr**idged (**abr**) and the raw RMSNet(-MiT)s, each of which applies its certain

Table 3.5: Ablation study on the effectiveness of BLSED strategy. "SAMixer-a" denote the two abridged SAMixers without BLSED strategy but applied with two different resolution reduction strategies of (1) a lightweight depth-wise convolutional layer (*i.e.*, *DW*), and (2) a heavyweight vanilla convolutional layer (*i.e.*, *V*), respectively.

Method		#Parameters	FPS	mIoU(%)↑
SegFormer [4]		27.36M	18.87	44.47
SAMixer-a	(DW)	30.34M	15.39	45.33
	(V)	32.69M	13.16	46.06
SAMixer		31.53M	16.81	**46.82**

Table 3.6: Ablation study on the resolution reduction ratio setting of the BLSED strategy.

Method	#Parameters	FPS	mIoU(%)↑
SegFormer [4]	27.36M	18.87	44.47
W/o (DW)	30.34M	15.39	45.33
$\mathbf{Ratio = 2}$	31.53M	16.81	**46.82**
$Ratio = 4$	32.15M	15.75	45.74

level(s) of feature(s) to perform RGB RMS, *i.e.*, from at least one **Lev**el (**Lv**) only (denoted by "(Abr-)RMSNet-Lv1" to "-Lv4") to the all four levels (*i.e.*, the raw RMSNet that is set as the base control group). *Note that the SAMixer module for multi-level feature fusion will be removed and replaced by a simple linear layer if performing RGB RMS with a single-level feature.* Our major observations are (1) each Abr-RMSNet that individually applies the single level of feature (*i.e.*, RMSNet-Lv1 to -Lv4) shows drastic accuracy drops compared to the raw RMSNet; (2) as for the intra-comparisons among RMSNet-Lv1 to -Lv4, (a) RMSNet-Lv1 demonstrates the relatively lowest mIoU and RMSNet-Lv2 outperforms RMSNet-Lv1 by a large margin; (b) RMSNet-Lv4 and -Lv3 yield comparable accuracies, yet RMSNet-Lv3 outperforms RMSNet-Lv4; (c) both RMSNet-Lv4 and -Lv3 outperform RMSNet-Lv2 and -Lv1 clearly, and (3) RMSNets demonstrate improving accuracies with the increasing of the number of feature levels. These results demonstrate the significance of the effective joint application of multi-level features from low to higher hierarchies for accurate RGB road scene material segmentation.

The collaboration of hierarchical feature encoding and adaptive feature fusion. First, it is expected that a weaker baseline model (*i.e.*, SegFormer-B1 [4] as for this ablation study) with lower mIoU is (likely) easier to be improved by the SAMixer module than the stronger counterpart (*i.e.*, SegFormer-B2 [4]) if taking into account the general marginal diminishing effect of accuracy and ignoring the inter-influence of the quality of feature encoding and fusion. The experimental results (shown in Table 3.8), however, demonstrate that

SAMixer brings even superior improvements to the stronger baseline SegFormer-B2 than the weaker baseline SegFormer-B1. This result validates the collaborative effect of hierarchical feature encoding and the MSA-based adaptive feature fusion, which also helps underscore our proposal of RMSNet framework that collaboratively leverages effective multi-level feature extraction and adaptive feature fusion.

Table 3.7: Experimental comparison results of RMSNets with different combinations of feature levels. "Level-1" to "Level-4" denote the corresponding "Levels" of feature, where "Level-4" is the highest level of feature which we apply as the anchor feature in the different feature combinations.

RMSNet	Level-4	Level-3	Level-2	Level-1	mIoU(%)↑
-Level-4 (Abr)	✓				38.88
-Level-3 (Abr)		✓			40.79
-Level-2 (Abr)			✓		33.65
-Level-1 (Abr)				✓	25.17
-Level-4,1 (Abr)	✓			✓	44.89
-Level-4,2,1 (Abr)	✓		✓	✓	46.23
-Level-4,3,2,1 (Raw)	✓	✓	✓	✓	**46.82**

Table 3.8: Accuracy improvements introduced by SAMixer module to the backbones MiT-B1 [4] and -B2 [4].

Encoder	Method	Decoder	#Params	mIoU(%)↑
Mix-Trs-B1 [4]	SegFormer [4]	ALL-MLP [4]	16.32M	42.27
	RMSNet (Ours)	**SAMixer (Ours)**	20.49M	**43.51**(+**1.24**)
Mix-Trs-B2 [4]	SegFormer [4]	ALL-MLP [4]	27.36M	44.47
	RMSNet (Ours)	**SAMixer (Ours)**	31.53M	**46.82**(+**2.35**)

Decoders for Ti-Feat Fusion

As discussed in Sec. 3.1, in semantic segmentation, Xie *et al*. [4] (*i.e*., SegFormer) demonstrated that SOTA segmentation heads with multi-level feature fusion (*e.g*., UperNet [77] and MLA (SETR) [78]) yielded inferior/close accuracies to the All-MLP decoder with Transformer-induced Features (Ti-Feats) which motivated us to introduce SAMixer.

We further validate our SAMixer for RGB RMS by comparing it with three other segmentation heads for multi-scale feature fusion, *i.e*., (1) ASPP (Atrous Spatial Pyramid Pooling) head of DeepLabv3+ [3]; (2) UperNet (Unified Perceptual Parsing Network) [77]; (3)

Table 3.9: Comparison of different decoders with Transformer-induced features. (1) "Trs" denotes "Transformer" and "-D" denotes "-Decoder."

Method	Encoder	Decoder	#Params	FPS	mIoU(%)↑
DeepLabv3+ [3]	ResNet101 [11]	ASPP-D [3]	59.34M	14.60	41.35
RN101-SAM (Ours)	ResNet101 [11]	**SAMixer (Ours)**	58.05M	15.48	**43.46**
SegFormer [4]	Mix-Trs-B2 [4]	ALL-MLP [4]	27.36M	18.87	44.47
MiT-B2-ASPP [3]	Mix-Trs-B2 [4]	ASPP-D [3]	29.63M	15.63	**43.88**
MiT-B2-UperNet [77]	Mix-Trs-B2 [4]	UperNet [77]	28.67M	17.41	45.38
MiT-B2-MLA [78]	Mix-Trs-B2 [4]	MLA [78]	29.21M	18.04	45.02
RMSN-MiT (Ours)	Mix-Trs-B2 [4]	**SAMixer (Ours)**	31.53M	16.81	**46.82**

MLA (Multi-Level feature Aggregation) head of SETR [78]. As a common property, these three popular multi-scale segmentation heads all learn feature pyramids with convolutions of varied receptive fields to mix features from different hierarchies.

We build three modified RMSNets by replacing our SAMixer with the decoders built on (1) ASPP, (2) UperNet, and (2) MLA, respectively. We add both SegFormer [4] (with the All-MLP decoder) and raw DeepLabv3+ as baselines. As shown in Table 3.9, with multi-level Ti-Feats extracted by the Mix-Transformer-B2 encoder, (1) ASPP head fails to improve the All-MLP decoder baseline; (2) UperNet and MLA improve accuracy over the All-MLP baseline but yields relatively marginal gains in accuracy. In contrast, SAMixer enjoys clear improvements over all the compared segmentation heads for multi-scale fusion and the baselines. This confirms the effectiveness of our SAMixer in fusion for informative texture and context cues across multi-level Ti-Feats for RGB RMS.

As an additional validation, we also demonstrate the effectiveness of our SAMixer for improving CNN features with MSA-based fusion. As shown in Table 3.9, by replacing the ASPP head with our SAMixer, the modified CNN framework based on the ResNet101 encoder outperforms the original DeepLabv3+ by a clear margin with comparatively small computational overhead.

3.5.4 Auxiliary Results

Evaluation with MCubeS

MCubeS [106] is a concurrent multimodal material segmentation dataset consisting of calibrated city scene images of different image modalities including RGB, NIR, and polarization images. We conduct RGB RMS evaluations with the color images of MCubeS dataset by comparing our RMSNet(s) with the methods that demonstrate competitive results on KITTI-Materials: (1) the popular CNN segmentation frameworks exploiting multi-scale feature encodings, *i.e.*, PSPNet [96], DeepLabv3 [98], and the encoder-decoder DeepLabv3+

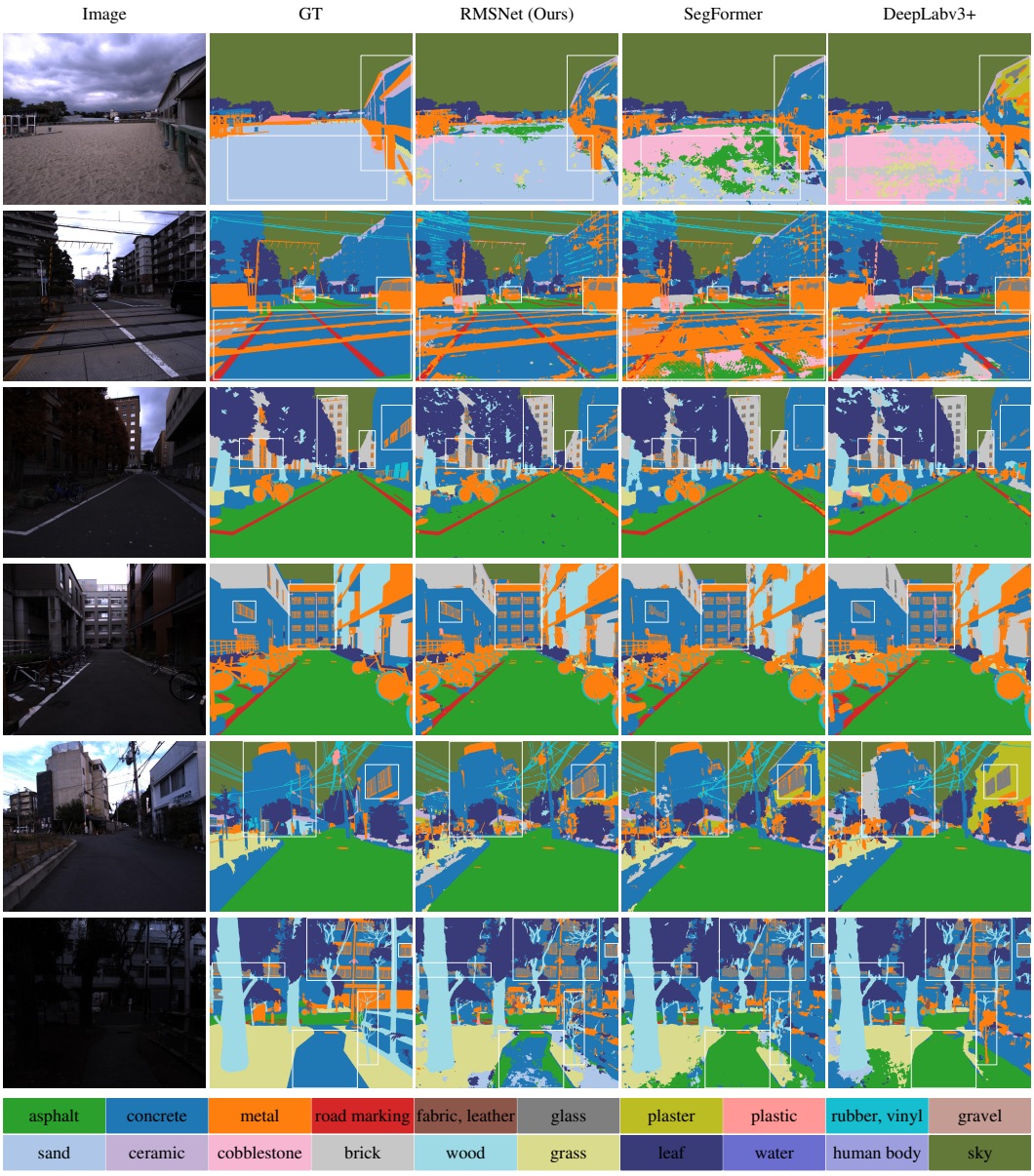


Figure 3.11: Example segmentation results on MCubeS for qualitative evaluations of our RMSNet(-MiT) and the related compared methods. RMSNet equipped with the SAMixer module for multi-level feature fusion achieves clearer segmentations on different materials of fragmented spatial distributions.

Table 3.10: Experimental comparison results on the MCubeS dataset. (1) "Trs" and "F" denotes "Transformer" and "Full," respectively. (2) "SEG" denotes "Segmentation." Methods with the suffix "-SEG" denote the segmentation networks built with the corresponding SOTA (MetaFormer) backbones (*i.e.*, ViT [19], CvT [20], DaViT [21], and ConvNeXt [22]) with the SOTA All-MLP decoder [4] that applies all the encoded levels of feature. "-D" denotes "-Decoder."

Method	Encoder	SEGHead	#Params	FPS	mIoU(%)↑
PSPNet (F) [96]	ResNet101 [11]	PSPNet [96]	65.58M	3.46	47.27
DeepLabv3 [98]	ResNet101 [11]	ASPP [98]	58.04M	3.89	45.41
DeepLabv3+ [3]	ResNet101 [11]	ASPP-D [3]	59.34M	4.04	47.70
ViT-SEG [19]	ViT-B/16 [19]	ALL-MLP [4]	89.03M	3.54	46.37
CvT-SEG [20]	CvT-13 [20]	ALL-MLP [4]	21.89M	4.66	47.97
SegFormer [4]	Mix-Trs-B2 [4]	ALL-MLP [4]	27.36M	5.09	48.37
RMSN-MiT (Ours)		**SAMixer (Ours)**	31.53M	4.75	**50.10**
DaViT-SEG [21]	DaViT-T [21]	ALL-MLP [4]	31.08M	5.25	47.53
RMSN-DVT (Ours)		**SAMixer (Ours)**	35.25M	4.86	**49.21**
ConvNeXt-SEG [22]	ConvNeXt-T [22]	ALL-MLP [4]	31.31M	5.39	49.89
RMSN-CNXt (Ours)		**SAMixer (Ours)**	35.48M	4.90	**51.67**

[3]; and (2) the related SOTA vision Transformers [19, 20, 4, 21]/ConvNeXt [22] using the SOTA All-MLP decoder [4] which merges multi-level features with linear aggregation, where the three SOTA backbones [4, 21, 22] are also applied in the corresponding RMSNets.

For robust evaluations of different RGB RMS encoder-decoder models, we adopt the suggested implementation protocols in Sec. 3.5.1, except for (1) prolonging the training epochs from 300 to 500 to exhaustively train the networks, as the training set of MCubeS includes relatively fewer RGB images than KITTI-Materials (302 vs. 800); (2) centrally cropping each image from 1024×1224 to 1024×1216 to ensure that the height and width of each image are both divisible by 16 (*i.e.*, a basic requirement for extracting features from four different hierarchies); and (3) expanding the test set from 102 (RGB) images to 198 images by merging the additional validation set into the raw test set. Table 3.10 reports the comparative results on MCubeS, where RMSNets equipped with SAMixer the novel MSA-based fusion module enjoy significant improvements over DeepLabv3+ the strong baseline, and other SOTA competitors. In particular, our SAMixer module demonstrates high consistency for the improvements on different SOTA hierarchical network backbones, which is in line with the evaluations on KITTI-Materials dataset.

Table 3.11: Per-class results on the MCubeS dataset. (1) "RMSN," "DVT," and "CNXt" denote "RMSNet," "DaViT," and "ConvNeXt," respectively. (2) Our RMSNet demonstrates clear improvements in IoU to the compared methods on varied materials including "asphalt," "concrete," "metal," "road marking," "glass," "plastic," "rubber, vinyl," "gravel," and "wood," which compose various significant objects of city scenes, including road surfaces, buildings, vehicles, bicycles, obstacles, and pedestrians.

Method	asphalt	concrete	metal	road marking	fabric, leather	glass	plaster	plastic	rubber, vinyl	sand	gravel	ceramic	cobblestone	brick	grass	wood	leaf	water	human body	sky	mean
PSPNet (F)	83.13	40.60	50.07	61.54	31.30	49.67	**4.68**	28.30	28.94	53.30	60.60	**36.30**	30.38	**32.62**	60.01	42.89	75.48	59.49	21.80	94.28	47.27
DeepLabv3	79.39	40.24	46.87	52.48	26.52	45.66	3.60	29.49	21.26	**61.45**	67.09	33.71	40.27	25.89	59.62	39.42	74.01	56.64	11.90	92.70	45.41
DeepLabv3+	80.25	44.53	50.58	64.95	32.91	50.91	3.82	29.43	33.77	44.63	59.11	31.72	35.66	27.10	61.19	44.76	76.59	67.50	18.68	95.86	47.70
ViT-SEG	79.94	42.81	45.26	64.66	23.88	42.55	3.04	21.19	29.55	56.81	53.19	32.21	**42.64**	26.44	62.47	39.91	75.02	**76.08**	13.66	95.99	46.37
CvT-SEG	80.42	44.19	49.22	63.32	32.40	48.05	3.62	26.21	32.25	55.54	56.63	33.20	27.48	29.62	**62.99**	42.39	**77.16**	73.77	**25.09**	95.84	47.97
SegFormer	81.88	43.94	50.54	62.84	**36.77**	50.29	3.96	28.35	34.22	51.07	63.16	35.53	33.94	29.84	59.76	44.82	76.92	65.14	18.34	**96.05**	48.37
RMSN-MiT	**84.42**	**46.64**	**50.59**	**67.62**	33.68	**51.67**	3.58	**29.85**	**35.85**	61.37	**67.94**	34.25	34.46	31.14	58.88	**45.39**	76.33	72.00	20.23	96.02	**50.10**
DaViT-SEG	80.02	42.83	50.06	64.49	29.11	48.46	**6.82**	26.84	33.39	49.85	53.04	35.48	**38.85**	27.78	57.38	44.62	76.61	64.76	**24.31**	**95.87**	47.53
RMSN-DVT	**83.54**	**46.20**	**51.22**	**64.93**	**30.68**	**50.79**	3.42	**29.98**	**35.63**	**60.06**	**61.49**	**37.09**	32.67	**31.01**	**60.97**	**45.16**	**77.11**	**67.69**	18.74	95.85	**49.21**
CNXt-SEG	**83.30**	42.94	51.25	66.92	32.95	51.83	5.46	29.97	34.84	56.55	52.65	38.81	**35.77**	27.93	63.47	45.14	77.38	71.24	33.40	96.05	49.89
RMSN-CNXt	82.44	**45.34**	**52.34**	**72.27**	**33.05**	**53.39**	**5.91**	**35.04**	**35.94**	**61.28**	**58.26**	**41.11**	32.78	**28.30**	**64.36**	**47.77**	**77.84**	**75.61**	**34.12**	**96.23**	**51.67**

We also report the per-class results for materials of our RMSNet(-MiT) and other competitor methods in Table 3.11 and qualitative visual segmentation examples in Figure 3.11, where RMSNet introduces clear accuracy gains on materials "asphalt," "concrete," "metal," "road marking," "glass," "plastic," "rubber, vinyl," "gravel," and "wood," which compose various significant objects of city scenes, including road surfaces, buildings, vehicles, bicycles, obstacles, and pedestrians. With the novel SAMixer model, our RMSNet variants RMSNet-DaViT and -ConvNeXt both enjoy significant improvements in IoU to DaViT-SEG and ConvNeXt-SEG with the SOTA All-MLP decoder on most material classes. This demonstrates the effectiveness of our framework RMSNet for RGB RMS.

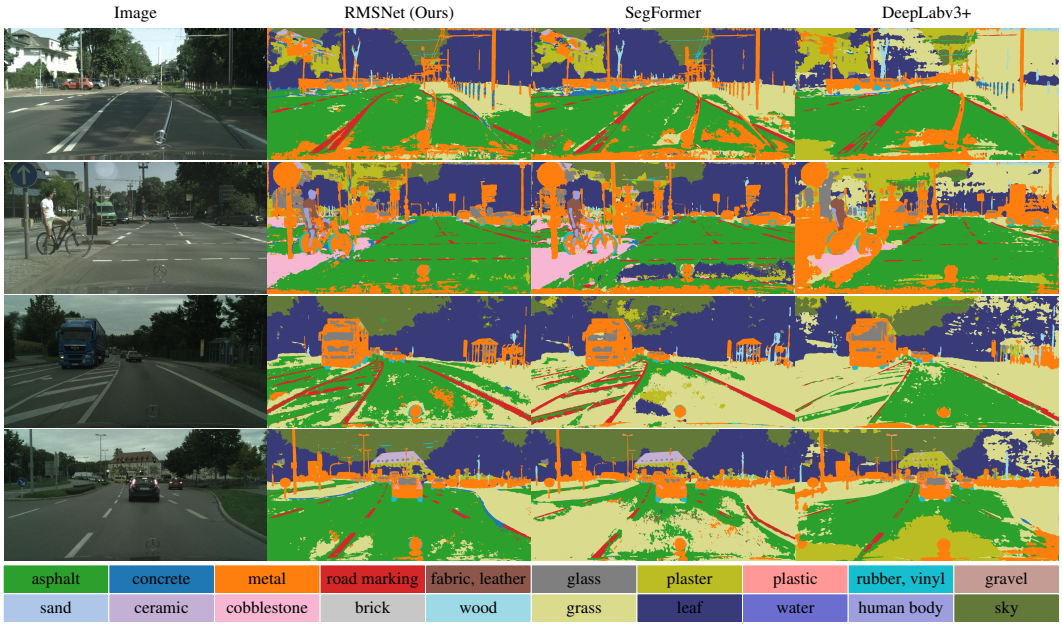


Figure 3.12: Visual examples on images from Cityscapes for qualitative evaluations.

Qualitative Results on Cityscapes

To further investigate the generalization ability of our RMSNet to realistic driving view images, we apply the RMSNet(-MiT) and compared models (*i.e.*, DeepLabv3+ [3] and SegFormer [4]) trained with KITTI-Materials to perform a per-pixel recognition on images from the validation set of Cityscapes [105] dataset (cities "Frankfurt" and "Lindau"). Note that we only show visual examples for qualitative evaluations due to the lack of corresponding material annotations in the Cityscapes dataset.

As shown in Figure 3.12, models trained with KITTI-Materials dataset are able to perform reasonable RGB RMS on images from Cityscapes. Furthermore, our RMSNet with the novel SAMixer achieves cleaner segmentation of different materials to the compared methods.

3.6 Summary

In this Chapter, Based on the benchmark dataset, KITTI-Materials, we address RGB road scene material segmentation, an emerging avenue of visual recognition problem, by deriving a new framework that effectively fuses texture and contextual cues. The framework, *i.e.*, RMSNet, achieves this with SAMixer, a novel model that performs effective yet efficient multi-level feature fusion with the tailored MSA mechanism, built on a newly derived balanced Q-K-Sim measure and BLSED strategy. Extensive experimental evaluations and ablation studies on KITTI-Materials dataset validate the effectiveness and scalability of our model designs.

Chapter 4

IIEU: Rethinking Neural Feature Activation from Decision-Making

4.1 Background

Nonlinear activation models that help fit the underlying mappings are one of the foundations for the unprecedented success of neural networks in pattern recognition tasks [30, 31, 32, 33]. The choice of the activation model is a decisive yet non-trivial factor in the performance of a neural network. Basic methods such as ReLU [5] and Softplus [38] are originated from neuronal behaviors [39, 40]. Based on them, past works have proposed to improve activation models with channel context (*e.g.*, FReLU [41], Dy-ReLU [42] and ACONs [7]), statistical strategies (*e.g.*, GELU [12], Pserf [15], and SMU [8]), and task-specific periodic functions [43, 44]. Existing methods, however, still leave critical problems in the optimal decision on activation models. As a major reason, although several past efforts [45, 46, 47] suggested extending activation models with dynamic approximators, it still lacks tailored hypotheses/interpretations to help specify the properties of effective activation models for pattern recognition. These specific properties, however, are difficult to be identified from pure biological intuitions.

To explore new improvements in feature activation, we rethink neural operations from MCDM (a typical problem in operational research) [122, 123, 124, 67, 125, 66, 126]. As a core of our MCDM hypothesis, we treat a nonlinear activation model as a selective re-calibrator that suppresses or emphasizes features according to their importance. Such importance, in fact, is first modeled by the feature-filter inner product which is supposed to

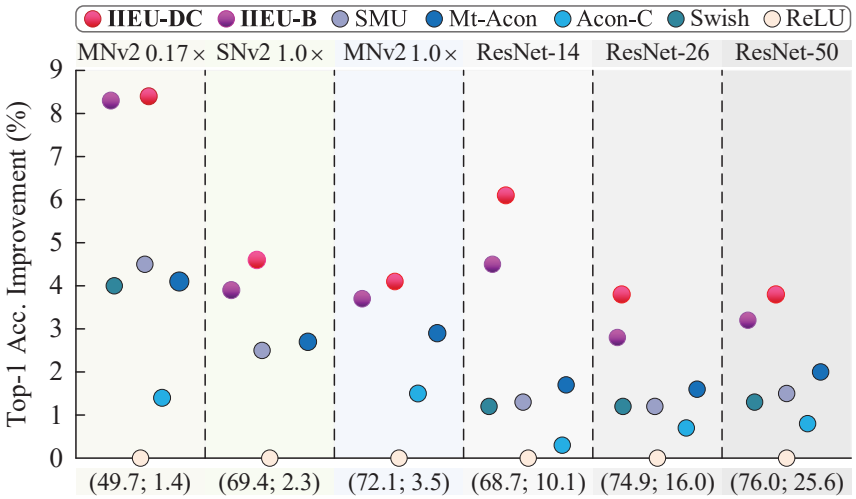


Figure 4.1: ImageNet Top-1 Accuracy (Acc.) relative improvements compared with the ReLU [5] baselines and SOTAs (Swish [6], ACONs [7] (CVPR'21), and SMU [8] (CVPR'22)) with (1) MobileNetV2 [9] (MNv2) 0.17× and 1.0×; (2) ShuffleNetV2 [10] (SNv2) 1.0×; (3) ResNet-14, -26, and 50 [11]. We show the ReLU baseline results by "(Acc.(%); parameters(M))". Our IIEUs achieve the new SOTA improvements to the ReLU baselines and outperform the SOTAs remarkably, with negligible/marginal additional parameters to ReLU (shown by the relative areas of the circular patterns, where each ReLU network denotes the unit area). ©2023 IEEE [2]

indicate the similarity of the feature to the filter. However, differentiated feature and filter norms can significantly bias the similarities modeled with feature-filter inner-products, thus likely interfering with the estimation of actual feature importance. We identify this as a critical yet unexcavated problem, namely *mismatched feature scoring* (as discussed in Figure 4.3(a)), which we infer from our hypothesis and otherwise invisible to past explanations.

To address the problem, we propose a set of specific properties of effective activation models with new intuitions and introduce the initial solution *i.e.*, a novel kind of activation models which we refer to as the *IIEUs*, to selectively re-calibrate features with an adaptive norm-decoupled importance measure. Specifically, we first treat each feature-filter inner product (suppose without biases and normalization layers) as a **T**ransitive **I**mportance (**TI**) score, as its input feature vector is de facto determined by a series of prior learning factors (*e.g.*, the initial input, the filters and activation models of the prior layers) and transmits their cues. We then estimate the corresponding norm-decoupled **I**nstantaneous **I**mportance (**II**) score with a low-cost adaptive shift term that incorporates mild learning adjustments. Finally, the feature activation is realized by multiplying each TI-score with the II-score. This feature re-calibration preserves meaningful prior learning information carried by the TI-scores yet eliminates the negative effect led by the *mismatched feature scoring* problem. Note that we formalize the *mismatched feature scoring* problem and TI-, II-scores in Section 4.3.

The main contributions of this work are 3-fold:

1. we suggest the MCDM hypothesis for neural feature activation, where we identify the unstudied yet critical problem of *mismatched feature scoring* in a typical neural activation process and introduce a set of new intuitions to help interpret the working mechanism of activation functions from a new generalized perspective of MCDM;
2. we introduce the novel activation prototype, IIEU, built from scratch on the suggested MCDM hypothesis, as the initial solution to the *mismatched feature scoring* problem;
3. we present the practical activation models, *i.e.* IIEU-B and IIEU-DC, based on the IIEU prototype and extensively validate (a) the effectiveness and versatility of IIEUs with various vision benchmark datasets, where IIEUs significantly improve the popular/SOTA activation functions; (b) our intuitions/hypothesis with targeted ablation studies.

4.2 Preliminaries

We consider the simple settings with image inputs:

(1) A network has T sequential learning layers indexed by $\tau = 1, 2, \ldots, T$. Let $\mathbf{X}^\tau \in \mathbb{R}^{C^\tau \times H^\tau \times L^\tau}$ which has C^τ channels and a spatial resolution of $H^\tau \times L^\tau$ denote the input feature map of the layer-τ.

(2) Let $x_c^{\tau+1}(h,l) := \phi\left(\tilde{x}_c^\tau(h,l)\right)$ denote the learning of the layer-τ at a given location $(h,l) \in \Omega_{H^\tau \times L^\tau}$ with the c-th filter $\boldsymbol{w}^\tau(c) \in \mathbb{R}^{C^\tau}$ and feature vector $\boldsymbol{x}^\tau(h,l) \in \mathbb{R}^{C^\tau}$, **where** $\tilde{x}_c^\tau(h,l) = \langle \boldsymbol{w}^\tau(c), \boldsymbol{x}^\tau(h,l) \rangle$ **denotes the inner product** and $\Omega_{H^\tau \times L^\tau}$ is the spatial lattice of $\mathbf{X}^\tau$. Note that the layer-τ includes a total of $C_{\tau+1}$ filters. $\phi : \mathbb{R} \to \mathbb{R}$ is a given activation function and we suppose $\phi\left(\tilde{x}_c^\tau(h,l)\right) = \rho\left(\tilde{x}_c^\tau(h,l)\right)\tilde{x}_c^\tau(h,l)$, where $\rho : \mathbb{R} \to \mathbb{R}$ defines the reweighting function of ϕ about $\tilde{x}_c^\tau(h,l)$.

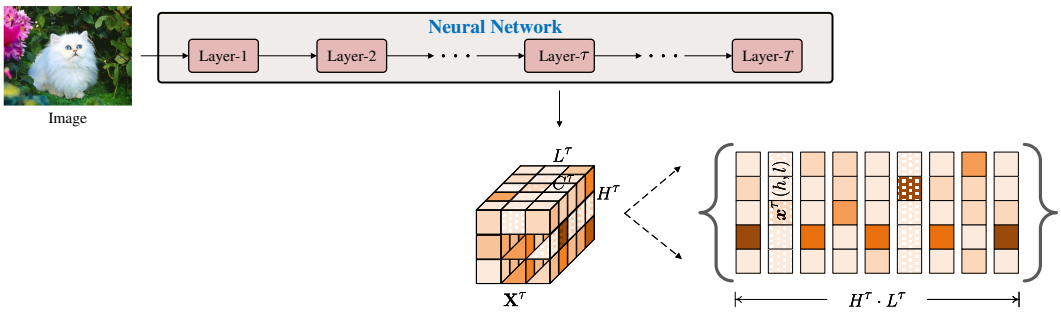


Figure 4.2: Intuitive illustration of the (sequential) network extractor, feature map $\mathbf{X}^\tau$ from layer-τ, and feature vector $\boldsymbol{x}^\tau(h,l)$ (on the feature map) in preliminary settings.

Note that **(1)** *we first leave aside normalization layers* (e.g., *BatchNorm (BN) [127] and LayerNorm (LN) [128]) and biases for simplicity and will consider them in* ***Section 4.3.2 (Practical Method)***. (2) for region-dependent learning with a $K \times K$ convolution, we meet the supposed settings by vectorizing the neighborhood of features/filters from size $C^\tau \times K \times K$ to $C^\tau \cdot K^2$. From MCDM, we treat (1) a filter $\boldsymbol{w}^\tau(c)$ as an updatable ideal

candidate[1] of the c-th group of criteria (*i.e.*, the C^τ channels of $\boldsymbol{w}^\tau(c)$); (2) a feature vector $\boldsymbol{x}^\tau(h,l)$ as an alternative candidate whose importance score about a group of criteria is measured by the feature-filter similarity, *i.e.*, **A**lternative-**I**deal (**A-I**) similarity. **Following we omit the layer index τ and spatial coordinate (h,l) to simplify the notations for the operations of layer-τ** (*e.g.*, we denote $\boldsymbol{x}^\tau(h,l)$, $\boldsymbol{w}^\tau(c)$, and $\tilde{x}_c^\tau(h,l)$ by $\boldsymbol{x}$, $\boldsymbol{w}$, and $\tilde{x}$, respectively). Therefore, a re-weighting-based neural feature activation process can be simplified as:

$$\phi(\tilde{x}) = \rho(\tilde{x}) \cdot \tilde{x}\,. \tag{4.1}$$

For clarity, **as for $\tilde{x} = \langle \boldsymbol{w}, \boldsymbol{x} \rangle$**, we simply use $\tilde{x}$ and $\boldsymbol{x}$ to mean the preliminary relatively influence of a candidate on the inferencing and filter updating, where the corresponding intensities are $|\tilde{x}|$ and $\|\boldsymbol{x}\|$, respectively, as in this case, (1) the difference of two vectorial candidates in a standard neural network can be measured by Euclidean distance; (2) a basic controlling factor of the influence of $\boldsymbol{x}$ on the updating of $\boldsymbol{w}$ is $\nabla_{\boldsymbol{w}} \langle \boldsymbol{w}, \boldsymbol{x} \rangle = \boldsymbol{x}$.

In particular, an extended case based on the assumed settings for discussing feature inference is: $\phi(\tilde{x}, \centerdot) = \rho(\tilde{x}, \centerdot)\,\tilde{x}$, $\phi, \rho : \dot{\mathbb{D}} \to \mathbb{R}$, where $\dot{\mathbb{D}}$ denotes the extended domain of $\tilde{x}$ with other given real variables/constants (denoted by $\centerdot$), if ϕ and ρ are still functions about $\tilde{x}$ when the values of other variables are known/fixed. In the following, we omit "$\centerdot$" (*e.g.*, denote $\phi(\tilde{x}, \centerdot)$ as $\phi(\tilde{x})$) if not specified. Note that an assumed extended condition of function is also considered for further generality: (1) for a discontinuous point on ρ, if it has the left-hand or/and right-hand limit(s) (but are unequal), let the single-side limit of the side where the point is defined by the (assumed-)limit of calculation; (2) for a non-differentiable point on ρ, if it has the left-hand or/and right-hand derivative, let the single-side derivative of the side where the point is defined by the (assumed-)derivative of calculation.

4.3 Rethinking Feature activation from MCDM

We aim to interpret neural feature activation from MCDM, find the unexplored critical problem, and propose our novel activation prototype and practical models by addressing the new problem. We first clarify our *Intuitions* and their induced *Properties*. We then present our IIEU prototype and practical methods constructed on them. For coherence, we discuss the related works in Section 4.4 with our hypothesis.

4.3.1 IIEU: Intuitions and Assumed Properties

In this subsection, we begin by rethinking the **meaning of neural feature activation from the perspective of MCDM** (**summarized by Intuition 4.1** and the discussion can be found in Appendix .2). We then introduce the ***mismatched feature scoring*** problem of neural activation newly inferred from our MCDM interpretation (**Intuition 4.3**). To investigate the

[1] With the given conditions, the ideal candidate in MCDM [67, 122, 124, 123] denotes the acquirable/virtual optimal choice capable of quantitatively measuring the performance of an alternative candidate by the similarity.

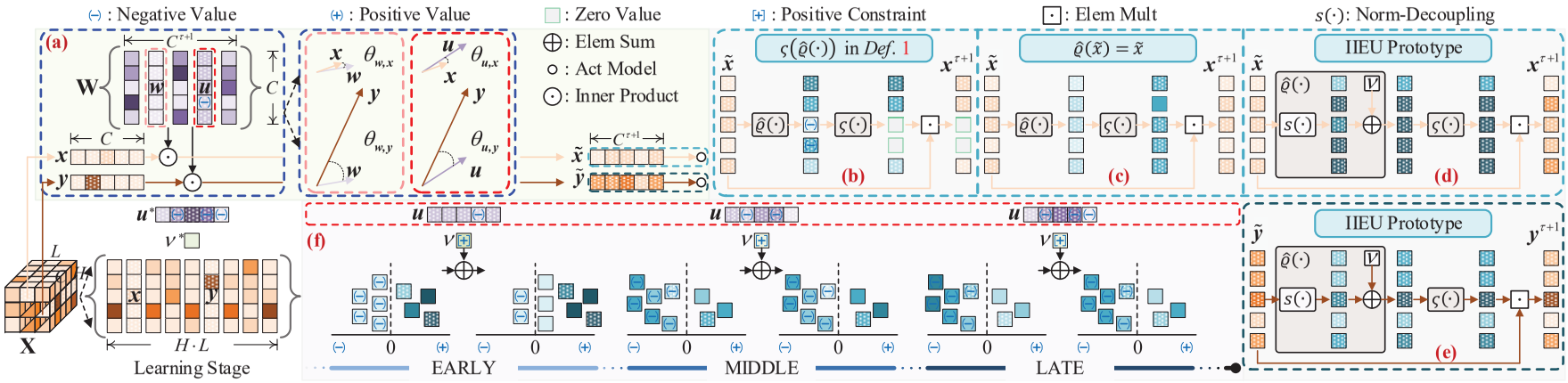


Figure 4.3: Illustration of the intuitions for IIEU. The shades of colors denote the intensities (the darker the higher and positive if w/o "(–)"), where "orange," "purple," "aqua," and "olive" denote features, filters, importance scores, and the parameters of the term-B. **(a)** *Mismatched feature scoring* problem: it is possible to find feature vectors $\boldsymbol{x}, \boldsymbol{y}$ and filters $\boldsymbol{w}, \boldsymbol{u}$ such that $\langle \boldsymbol{u}, \boldsymbol{y} \rangle \gg \langle \boldsymbol{w}, \boldsymbol{x} \rangle$ and $\langle \boldsymbol{w}, \boldsymbol{y} \rangle \gg \langle \boldsymbol{w}, \boldsymbol{x} \rangle$, where $\boldsymbol{y}$ is far dissimilar to $\boldsymbol{u}$ and $\boldsymbol{w}$ compared with $\boldsymbol{x}$ to $\boldsymbol{w}$, due to the significant differences of the norms. **(b)** Intuition 4.1: a "nonlinear" activation model does not be specified to suppress/emphasize candidates with their expected importance. **(c)** An example of typical activation model, where $\tilde{x}$ is directly applied as the approximated similarity $\hat{\varrho}(\tilde{x})$ and the **(a)** is left unsolved. **(d)** and **(e)** IIEU eliminates the **(a)** by scoring feature with the adaptive norm-decoupled approximated similarity, such that the influence of $\boldsymbol{x}$ are relatively emphasized by assigning higher scores compared to $\boldsymbol{y}$. **(f)** *Properties of the term*-B: $\boldsymbol{u}^*, \boldsymbol{\nu}^*$ denote the (virtual) optimal $\boldsymbol{u}, \boldsymbol{\nu}$ for $\boldsymbol{u}, \boldsymbol{\nu}$ to approach in training, respectively. we suppose ν to be updatable, positive, and bounded since (1) the perfectness of filters as ideal candidates cannot be ensured (as discussed with Intuition 4.3); (2) we identify the positive translation to the codomain of the approximated similarity $\hat{\varrho}(\tilde{x})$ help to selectively suppress/emphasize the influence of targeted candidates; (3) a bounded ν ensures that the contribution of the bounded main term-S will not be neutralized by the auxiliary ν (as further discussed in Section 4.3.2 with the ablation study (4)). ©2023 IEEE [2]

solution to the *mismatched feature scoring* problem (*i.e.*, the novel class of neural activation model, **IIEU(s)**), we first decompose the basic self-gated neural activation process (Equation (4.1)) to a more specific MCDM-inspired process (Equation (4.2)) and then present **four new intuitions** (*i.e.*, Intuitions 4.2, 4.4, 4.5, and 4.6) that inspire the **four qualitative properties** (*i.e.*, Properties 4.1, 4.2, 4.3, and 4.4) of neural feature activation for image-based visual recognition, which help specify the design of our novel IIEU(s) from scratch.

"Nonlinearity" is indispensable for the learning of discriminative neural representations. The mathematically absolute "nonlinearity," however, can also be brought by other basic operations, *e.g.*, BN [127], LN [128], and the biases of linear layers. From MCDM, as for an activation model, non-important candidates are likely to be scored with negative A-I inner products, where the candidates with intense negative inner products are possible to deteriorate the learning. This necessitates a *selective* re-calibration to suppress/preserve the harmful/positive influence of input features, respectively. Our Intuition 4.1 aims to re-interpret the meaning of neural feature activation from MCDM and stems our following intuitions and qualitative properties for effective neural activation.

Intuition 4.1. From MCDM, we regard neural feature activation as a significant re-calibrator to cast *selective* re-calibrations on input features (to the activation model) to suppress and/or preserve the harmful and/or positive influence of the corresponding features according to the (measured) importance scores of features, respectively.

As the importance scores of features (mentioned in Intuition 4.1) are modeled based on the feature-filter similarities, for clearer discussion, we decompose the re-weighting function $\rho\left(\tilde{x}\right)$ of basic self-gated neural activation (Equation (4.1)) to a more specific MCDM-inspired process, *i.e.*,

$$\rho\left(\tilde{x}\right) = \varsigma\left(\varrho\left(\tilde{x}\right)\right) \ , \tag{4.2}$$

where $\varrho\left(\tilde{x}\right)$ is assumed to be an ideal (*i.e.*, unbiased) similarity measure capable of measuring the unbiased similarity of the input feature vector $\boldsymbol{x}$ to the filter $\boldsymbol{w}$ by rectifying the feature-filter inner product $\tilde{x}$; $\varsigma\left(\cdot\right)$ is an ***adjuster*** function that casts suitable constraints on the codomain of ϱ. In particular, we can specify the core attribute of the *ideal similarity* through Intuition 4.2.

Intuition 4.2. For any given alternative and ideal candidates $\boldsymbol{x}, \boldsymbol{y}$ and $\boldsymbol{w}, \boldsymbol{v}$, suppose $\tilde{x} = \langle \boldsymbol{w}, \boldsymbol{x} \rangle$ and $\tilde{y} = \langle \boldsymbol{v}, \boldsymbol{y} \rangle$. If $\varrho\left(\tilde{x}\right) \geq \varrho\left(\tilde{y}\right)$, then, $\boldsymbol{x}$ has higher/equal importance than $\boldsymbol{y}$ about their importance measure criteria.

Then, the function of the adjuster $\varsigma\left(\cdot\right)$ on the unbiased similarity $\varrho\left(\tilde{x}\right)$ can be specified by the assumed Property 4.1.

Property 4.1. $\left|\varsigma\left(\varrho\left(\tilde{x}\right)\right)\right| \geq \left|\varsigma\left(\varrho\left(\tilde{y}\right)\right)\right|$ if $\varrho\left(\tilde{x}\right) \geq \varrho\left(\tilde{y}\right)$.

In particular, we assumed that $\varsigma\left(\varrho\left(\tilde{x}\right)\right)$ is continuous and differentiable about $\varrho\left(\tilde{x}\right)$ on the domain (or at most has finite points where the left- and right-hand limits of the function

exist but are unequal). In fact, the adjuster ς can be treated as a rectification applied to ensure the monotonicity of the absolute importance score of $\boldsymbol{x}$ to $\boldsymbol{w}$, *i.e.* $|\rho(\tilde{x})|$ (the absolute weight) about $\varrho(\tilde{x})$, since we clarified that the influence of a feature to the update of a filter is mainly controlled by the absolute intensities of the elements of feature. Further, we identify the property of the adjuster ς can be ensured by the simple condition clarified in Proposition 4.1.

Proposition 4.1. *Property 4.1 $\Longleftrightarrow$ (1) $\varsigma(\varrho_x)$ is (monotonically) non-decreasing about $\varrho_x \wedge \varsigma(\varrho_x) \geq 0 \vee$ (2) $\varsigma(\varrho_x)$ is (monotonically) non-increasing about $\varrho_x \wedge \varsigma(\varrho_x) \leq 0$ (ϱ_x denotes $\varrho(\tilde{x})$; $\wedge$ denotes "and;" $\vee$ denotes "or").*

In particular, for the design of practical activation models, we discuss $\varsigma(\varrho(\tilde{x})) \geq 0$ (*i.e.*, $\varsigma(\varrho(\tilde{x}))$ is lower-bounded) without loss of generality. Then, as the underlying mappings of neural learning are usually extremely complex, we assume that the direct defining of the ideal similarity $\varrho(\tilde{x})$ can be excessively difficult. For practical application, we instead propose to learn to approximate $\varrho(\tilde{x})$ by a flexible and updatable similarity measure, $\hat{\varrho}(\tilde{x})$, which we refer to as the **rectified (approximated) similarity**. Therefore, the core idea of our activation function IIEU(s) is introducing a novel norm-decoupled *rectified similarity* (Equation (4.3)) to address the ***mismatched feature scoring*** problem clarified by Intuition 4.3.

Intuition 4.3. **Mismatched feature scoring**. Large feature norms and filter norms possibly significantly bias the inner-product-based feature-filter similarities, hence taking away from how important the features actually are. We refer to this problem as the *Mismatched feature scoring* of neural feature activation.

An example is illustrated in Figure 4.3(a) to help clarify this intuition. Note that typical activation functions commonly apply feature-filter inner products $\tilde{x}$ as the rectified similarities, *i.e.*, $\hat{\varrho}(\tilde{x}) = \tilde{x}$ (omitting normalization layers and biases, as illustrated in Figure 4.3(c)). However, as discussed in Intuition 4.3, this possibly leads to unreliable similarity measure for features to the concerned filters.

More specifically, the reasons why we suppose that norms of features and filters possibly introduce distracting biases are two-fold:

(1) The norm of a feature (vector), *i.e.* $\|\boldsymbol{x}\|$, is actually determined by the prior learning layers (*i.e.*, layer-1 to layer-$(\tau-1)$) and the initialization yet having a weak relationship to the current learning layer (*i.e.*, layer-τ).

(2) The inter-filter relationships in a learning layer (layer-τ) are relaxed (*i.e.*, without specific modeling in neural operations), so the norms of filters $\|\boldsymbol{w}\|$ in the feature-filter inner product may be hard to compare in the meaning of feature-filter similarities.

More intuitively, these demonstrate that feature norms and filter norms possibly bring significant *impurities* about the measuring of current similarities of the given feature to the concerned filter, especially when we compare cross-input similarities (*e.g.*, if comparing $\tilde{x} = \langle \boldsymbol{w}, \boldsymbol{x} \rangle$ to $\tilde{y} = \langle \boldsymbol{u}, \boldsymbol{y} \rangle$, as illustrated in Figure 4.3(a)).

Especially, based on this intuition, we identify $\cos\theta_{\boldsymbol{w},\boldsymbol{x}} = \frac{\tilde{x}}{\|\boldsymbol{w}\|\|\boldsymbol{x}\|}$, *i.e.*, the cosine similarity a suitable measure to reflect the feature-filter similarity, because it focuses on the current information of the feature and filter. Moreover, if taking into account the uses of normalization layers and/or biases for linear projections (denoted by ψ), a modified (generalized) similarity can be $\tilde{x} = \psi\left(\langle \boldsymbol{w}, \boldsymbol{x} \rangle\right)$ which alleviates the influence of norms by explicit norm-decoupling. Based on this idea, we propose ***IIEU*** prototype as the first solution to address the *mismatched feature scoring* problem, *i.e.*:

$$\phi\left(\tilde{x}\right) = \varsigma\left(\frac{\tilde{x}}{\|\boldsymbol{x}\|\,\|\boldsymbol{w}\|} + \nu\right) \cdot \tilde{x}\,. \tag{4.3}$$

Note that figures 4.3(d) and 4.3(e) illustrate the (intuitive) operations of IIEU. Here, we let $\hat{\varrho}\left(\tilde{x}\right) = \frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|} + \nu$ as the rectified similarity to address *mismatched feature scoring*, where we introduce an updatable bias term ν (denoted by term-*B*, the auxiliary bias term) to incorporate further learning flexibility to rectify the modified (cosine) similarity term $\frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|}$ (denoted by term-*S*, *i.e.*, the main similarity term). Then, ς is a tailored-made adjuster function to ensure the satisfactions of the **four assumed properties** of the overall importance scoring (*i.e.*, the assigned weight $\rho\left(\tilde{x}\right) = \varsigma\left(\frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|} + \nu\right)$), which we refer to as the **I**nstantaneous **I**mportance score of IIEU. That is, IIEU realizes norm-decoupled feature activation by rectifying (*i.e.*, multiplying) each biased comprehensive score $\tilde{x}$ (*i.e.*, the relaxed feature-filter similarity) with its corresponding estimated II-score, respectively. **Note that besides Property 4.1, the other three properties (CNI 4.2, PPI 4.3, and OD 4.4) are introduced in the following text.**

More specifically, a critical reason why we introduce ν is that filters, the ideal candidates, are first assumed to be perfect representatives of the certain combinations of channels (*i.e.*, the criteria). However, the perfectness of filters, in fact, can hardly satisfy in practical applications, especially in the early and medium training stages where filters are far from being optimized. This necessitates flexible (learnable) rectifications to term-*S*. Note that we consider specific constraints on ν (*e.g.*, positive constraint), which are illustrated and discussed in Figure 4.3(f).

Further, we assume $\|\boldsymbol{x}\|\,\|\boldsymbol{w}\| > 0$. In practical application, we add a small value (*e.g.*, 10^{-6}) to $\|\boldsymbol{x}\|\,\|\boldsymbol{w}\|$ to prevent the zero denominators, which also ensures term-*S* to be bounded (discussed in the following paragraph). Note that IIEU prototype (Equation (4.3)) possibly becomes a non-function mapping about $\tilde{x}$ yet we can treat and analyze it as a function when discussing feature inference because (1) a neural network has a finite (limited) number of filters and input images (especially in a mini-batch), where the number of filters and input features of a single layer (*e.g.*, layer-τ) is constrained (*i.e.*, not so many); (2) the computer calculates neural operations in a 32-bit space with (pseudo-)continuous values. **Therefore, for filters $\boldsymbol{w}, \boldsymbol{v}$ and input features $\boldsymbol{x}, \boldsymbol{y}$ of a layer-τ, where $\boldsymbol{w} \neq \boldsymbol{v} \vee \boldsymbol{x} \neq \boldsymbol{y}$, the chance to have $\langle \boldsymbol{w}, \boldsymbol{x} \rangle = \langle \boldsymbol{v}, \boldsymbol{y} \rangle$ can be extremely low (*i.e.*, negligible)** (the detailed discussion is included in the Appendix .1.3).

Below, we introduce the other three assumed properties ***C**onstraint on **N**egative **I**nfluence (**CNI**, 4.2)*, ***P**reservation on **P**ositive **I**nfluence (**PPI**, 4.3)*, and ***O**riented **D**iscriminativeness (**OD**, 4.4)*, which further specify the relationships of the adjuster $\varsigma(\cdot)$ and the rectified similarity $\hat{\varrho}(\tilde{x})$ for neural feature activation based on MCDM interpretation.

Intuition 4.4. ***CNI***: Any negative candidate is expected to be assigned with a limited weight to constrain its influence.

Intuition 4.5. ***PPI***: Any two important candidates with close importance are expected to be assigned with comparable weights to ensure comparable influence, *i.e.*, the relatively more important candidate will not cover the influence of another.

Intuition 4.6. ***OD***: The assigned weights are expected to differentiate the positive and the negative candidates (while avoiding gradient and feature vanishing or explosion).

These three intuitions introduce dependent constraints on the influence of negative and positive candidates (*i.e.*, features). More concretely, we suggest three assumed properties based on corresponding intuitions, respectively, as follows.

Property 4.2. ***(CNI)*** (Basic case:) $\exists \eta \in \mathbb{R}, \mathcal{M}_{x^-} \geq 0$ such that $\forall \varrho(\tilde{x}) < \eta$, we have $|\varsigma(\varrho(\tilde{x}))| \mid_{\varrho(\tilde{x})<\eta} \leq \mathcal{M}_{x^-}$, especially, $\lim_{\varrho(\tilde{x})\to-\infty} |\varsigma(\varrho(\tilde{x}))| = 0$.

Property 4.3. ***(PPI)*** (Basic case:) $\exists \eta \in \mathbb{R}, \mathcal{M}_{x^+} \geq 0$ such that $\forall \varrho(\tilde{x}) > \eta$ we have $|\varsigma(\varrho(\tilde{x}))| \mid_{\varrho(\tilde{x})>\eta} \leq \mathcal{M}_{x^+}$.

Property 4.4. ***(OD)*** $\exists \eta \in \mathbb{R}$ and $\exists \epsilon_\rho, \delta_\rho > 0$ such that if $\varrho(\tilde{x}) > \eta > \varrho(\tilde{y})$, then, $\forall \varrho(\tilde{x}) - \varrho(\tilde{y}) > \epsilon_\rho$ we have $\varsigma(\varrho(\tilde{x})) - \varsigma(\varrho(\tilde{y})) > \delta_\rho$. Note that δ_ρ is assumed to be an appropriate value to avoid gradient and feature vanishing or explosion.

Note that we also introduce the strict cases of Property 4.2 and Property 4.3 in Appendix .1.2, respectively, for where $\varrho(\tilde{x})$ is (a) (uniformly) continuous about $\tilde{x}$ on the domain; (b) differentiable about $\tilde{x}$ or at most has a finite number of points where the left- and right-hand limits exist but are unequal. These underlie our new work in Chapter 5.

By leveraging the four assumed properties (*i.e.*, Properties 4.1, 4.2, 4.3, and 4.4) and the Intuition 4.3, following we present **IIEU-B** and **IIEU-DC** as two practical activation models of IIEU. In particular, as II-score built upon the proposed approximation to the ideal similarity (*i.e.*, $\hat{\varrho}(\cdot)$), we suppose a relaxed Property 4.1 to bring additional learning flexibility, *i.e.*, $\varsigma(\hat{\varrho}(\tilde{x}))$ is possible to have small negative values and be non-monotonic about $|\hat{\varrho}(\tilde{x})|$ at $|\hat{\varrho}(\tilde{x})| \leq |\eta|$, where η denotes a given threshold close to 0 (*i.e.*, Equation (4.5)).

Especially, we suppose the differentiability of ϕ about $\tilde{x}$ is not a necessary constraint for neural activation, because it does not be involved in the gradient computing and filter updating process. In contrast, for enabling effective gradient computing, we expect an activation model to be continuous and differentiable (or at most has finite points where the left- and right-hand limits of the function exist but are unequal) about $\boldsymbol{w}$, *i.e.*, the filters. This constraint is applicable to our practical IIEUs (some relevant calculations, *e.g.*, Equations (4.7) and (4.8) are included in Section 4.3.2).

Moreover, we experimentally validate the effectiveness of our MCDM hypothesis (the new intuitions and suggested properties) for qualitative preliminary assessments of neural activation models (for image-based visual recognition) in Appendix .6 (with AdaShift, *i.e.*, the novel neural activation model(s) we newly propose with improved practical efficiency and generalization ability to IIEU(s)).

4.3.2 Practical Method

We present IIEU-**B** (**Basic**, Figure 4.4) as the initial practical IIEU and IIEU-**DC** (**D**ynamic **C**oupler, Figure 4.5) as a tailored enhancement to IIEU-B. In the subsequent, we introduce IIEU-B and IIEU-DC in detail.

Formulation.

We propose IIEU-B built on the prototype of IIEU (Equation (4.3)) described in Section 4.3.1, by embodying the term-*B* ν and the adjuster ς with the proposed properties. Specifically, for IIEU-B, we let term-*B* be

$$\nu = \delta\left(\mathrm{LN}\left(\mathrm{avgpool}\left(\tilde{\boldsymbol{X}}_c\right)\right)\right), \tag{4.4}$$

where LN denotes the LayerNorm [128] to perform flexible channel-dependent scaling and shift to channel statistics with negligible cost. δ is Sigmoid function to cast upper-bounded positive constraint on channel statistics to help meet the supposed properties (Figure 4.3(f)). Moreover, with the prior that $\hat{\varrho}\left(\tilde{x}\right)$ of IIEU-B is bounded, we propose a suitable conditional adjuster ς to meet the proposed Properties:

$$\varsigma\left(\hat{\varrho}_x\right) = \begin{cases} \hat{\varrho}_x\,, & \hat{\varrho}_x \geq \eta\,; \\ \eta \exp\left(\hat{\varrho}_x - \eta\right), & \hat{\varrho}_x < \eta\,; \end{cases} \tag{4.5}$$

where $\hat{\varrho}_x$ denotes $\hat{\varrho}\left(\tilde{x}\right)$ and η a learnable threshold shared within each channel, initialized by a small value (0.05 by default). Note that (1) $\varsigma\left(\hat{\varrho}_x\right)$ is (uniformly) continuous about $\hat{\varrho}_x$ on the domain as its non-differentiable point ($\hat{\varrho}_x = \eta$) possesses $\lim_{\hat{\varrho}_x - \eta \to 0^-} = \lim_{\hat{\varrho}_x - \eta \to 0^+} = \eta$; (2) we suppose the right-hand derivative as the derivative at $\hat{\varrho}_x = \eta$ (Section 4.2); (3) the influence of any candidate with $\hat{\varrho}_x \leq \eta$ will be silenced if $\eta = 0$.

Boundedness of $\rho\left(\tilde{x}\right)$.

We suppose the boundedness of II-score $\rho\left(\tilde{x}\right)$ as a significant condition to ensure training stability. As for IIEU-B, as ν is bounded and ς is conditionally linear about $\hat{\varrho}_x$ for $\hat{\varrho}_x > \eta$, the boundedness of $\rho\left(\tilde{x}\right)$ is solely determined by the term-*S*. For generality, we discuss the common case that BatchNorm [127] is applied, *i.e.*, with the channel scaling and shift factors $\gamma, \beta \in \mathbb{R}$ (extensible to LayerNorm [128]). Let $E = \|\boldsymbol{x}\|\,\|\boldsymbol{w}\| \neq 0$, the codomain of term-*S* is calculated as (calculation details can be found in Appendix .3.1):

$$-|r| + \frac{\beta - r\mu}{E} \leq \frac{\tilde{x}}{E} \leq |r| + \frac{\beta - r\mu}{E}, \tag{4.6}$$

where $r = \frac{\gamma}{\sigma}$; $\sigma \neq 0$ and μ denote the standard deviation and mean of $\tilde{x}$ for channel-c. That is, we can calculate both the upper- and lower-bound of term-*S* with the factors γ and β whose values are constrained by the weight-decay (*i.e.*, $\mathcal{L}_2$-regularization) in the training phase. Unlike the cosine similarity with a range $[0, 1]$, the range of term-*S* can be broader. Moreover, as the adjuster ς constrains $\rho(\hat{\varrho}_x) < \eta$ for $\hat{\varrho}_x < \eta$, the II-score can adaptively emphasize/suppress the informative/meaningless candidates.

Comparative discussion of term-*S* and -*B* .

We suppose term-*B* to be bounded to prevent it from neutralizing the contribution of the term-*S* (Figure 4.3(f)). In this paragraph, **we first discuss the case that** $\hat{\varrho}_x \geq \eta$**, *i.e.*,** $\varsigma(\hat{\varrho}_x) = \hat{\varrho}_x$ **such that** $\phi(\tilde{x}) = \varsigma\left(\frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|} + \nu\right)\tilde{x} = \frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|}\tilde{x} + \nu\tilde{x}$. Further, we simplify the case of comparing term-*B* and -*S* by considering $\tilde{x} = \langle \boldsymbol{w}, \boldsymbol{x} \rangle$ without loss of generality, as they share the same BN layers. Note that we discuss the derivatives about $\boldsymbol{w}$ and denote the term-*S* and -*B* by $s(\boldsymbol{w}) = \frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|}$ and $\nu(\boldsymbol{w}) = \delta\left(\mathrm{LN}\left(\bar{\tilde{x}}\right)\right)$, respectively, where $\bar{\tilde{x}}$ denotes the mean statistic for channel-c and δ is the Sigmoid function. Moreover, we approximate the operation of LN by $\mathrm{LN}\left(\bar{\tilde{x}}\right) = \dot{\gamma}\left(\bar{\tilde{x}}\right) + \dot{\beta}$, where $\dot{\gamma}$ and $\dot{\beta}$ are the scaling and shift factors of the LN layer. Then, we can calculate the (partial) derivative about $\boldsymbol{w}$ of $s(\boldsymbol{w})$ as:

$$\nabla_{\boldsymbol{w}} s(\boldsymbol{w}) = \frac{\|\boldsymbol{w}\|^2 \boldsymbol{x} - \boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}}{\|\boldsymbol{x}\| \|\boldsymbol{w}\|^3}, \tag{4.7}$$

where T denotes matrix/vector transpose (calculation details can be found in Appendix .3.2). Correspondingly, we calculate the derivative about $\boldsymbol{w}$ for term-*B* as (Appendix .3.3):

$$\nabla_{\boldsymbol{w}} \nu(\boldsymbol{w}) = \delta\left(\dot{\gamma}\boldsymbol{w}^{\mathrm{T}}\bar{\boldsymbol{x}} + \dot{\beta}\right)\left(1 - \delta\left(\dot{\gamma}\boldsymbol{w}^{\mathrm{T}}\bar{\boldsymbol{x}} + \dot{\beta}\right)\right)\dot{\gamma}\bar{\boldsymbol{x}}, \tag{4.8}$$

where $\bar{\boldsymbol{x}} = \mathrm{avgpool}(\mathbf{X}) \in \mathbb{R}^C$ denotes the vectorial channel mean statistics of the feature map $\mathbf{X}$. Particularly, we can expand the top-right term in Equation (4.7) as:

$$\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} = \left(\sum_{c=1}^{C} w_c x_c\right)\boldsymbol{w}. \tag{4.9}$$

That is, we identify term-*S* enabling each neuron to model detailed cross-channel feature-filter interactions at every spatial coordinate and leverage these informative cues to improve the filter updating (calculation details can be found in Appendix .3.4). In contrast, as a control group, we calculate the derivative about $\boldsymbol{w}$ of ReLU [5] as:

$$\nabla_{\boldsymbol{w}} \mathrm{ReLU}(\tilde{x}) \,|_{\langle \boldsymbol{w}, \boldsymbol{x} \rangle > 0} = \boldsymbol{x}, \tag{4.10}$$

where ReLU is shown to model channel-independent information only and lacks the capability to improve filter updating with inter-channel relationships.

Next, we discuss the function of term-*B* from filter updating, which de facto realizes aligned adaptive adjustments to the term-*S* with statistical inter-channel information. As the

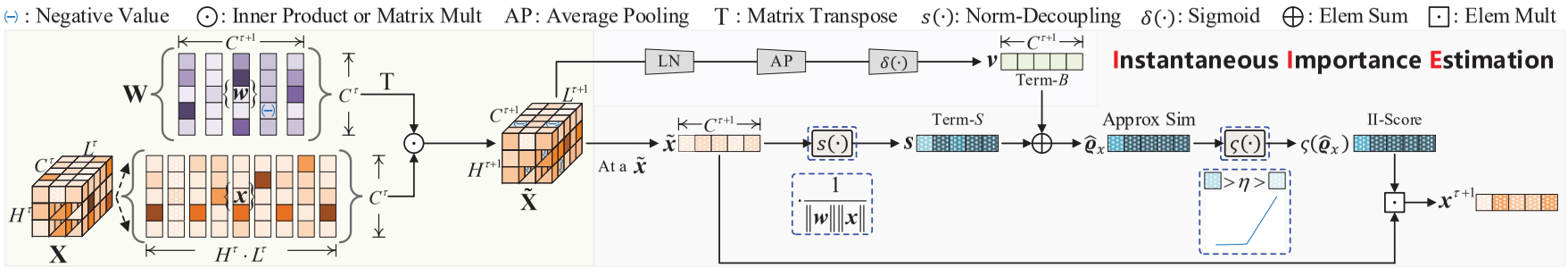


Figure 4.4: Operational illustration of IIEU-B. “Elem” and “Mult” denote “Element-wise” and “Multiplication,” respectively. ©2023 IEEE [2]

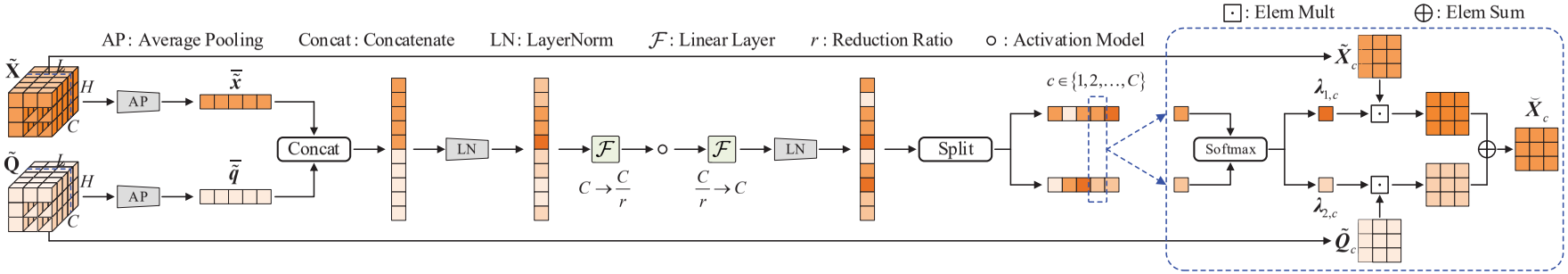


Figure 4.5: Operational illustration of Dynamic Coupler module. $\bar{\tilde{x}}, \bar{\tilde{q}} \in \mathbb{R}$ denote the vectorial channel statistics of the main branch feature map $\tilde{\mathbf{X}}$ and the residual feature map $\tilde{\mathbf{Q}}$. ©2023 IEEE [2]

representative of long-range channel cues, term-*B*, however, does not provide the instance details about the features, hence it may dilute the contribution of the term-*S* if having excessive derivative about the filter. We propose to eliminate this problem by casting a positive constraint on the term-*B* with Sigmoid. Specifically, as the terms $\|\boldsymbol{w}\|^2 \boldsymbol{x}$ and $\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$ in Equation (4.7) are composed of the same member vectors (*i.e.*, $\boldsymbol{w}$ and $\boldsymbol{x}$), without loss of generality, we suppose $\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} = -\alpha \|\boldsymbol{w}\|^2 \boldsymbol{x}, \alpha \in \mathbb{R}$ and we have

$$\nabla_{\boldsymbol{w}} s(\boldsymbol{w}) = \frac{\|\boldsymbol{w}\|^2 \boldsymbol{x} + \alpha \|\boldsymbol{w}\|^2 \boldsymbol{x}}{\|\boldsymbol{x}\| \|\boldsymbol{w}\|^3} = \frac{1+\alpha}{\|\boldsymbol{x}\| \|\boldsymbol{w}\|} \boldsymbol{x} . \tag{4.11}$$

Then, we can calculate the average contribution of term-*S* to the updating of filter $\boldsymbol{w}$ as:

$$\left|\nabla_{\boldsymbol{w}} \bar{s}(\boldsymbol{w})\right| = \left|\frac{1+\alpha}{\|\bar{\boldsymbol{x}}\| \|\boldsymbol{w}\|}\right| |\bar{\boldsymbol{x}}| . \tag{4.12}$$

With the preceding conditions, we have a conditional corollary:

$$\left|\frac{1+\alpha}{\|\bar{\boldsymbol{x}}\| \|\boldsymbol{w}\|}\right| \geq \frac{1}{4} |\dot{\gamma}| \Longrightarrow \left|\nabla_{\boldsymbol{w}} \bar{s}(\boldsymbol{w})\right| \geq |\nabla_{\boldsymbol{w}} \nu(\boldsymbol{w})| , \tag{4.13}$$

because (calculation details can be found in Appendix .3.5)

$$\nabla_{\boldsymbol{w}} |\nu(\boldsymbol{w})| \leq \frac{1}{4} |\dot{\gamma}| |\bar{\boldsymbol{x}}| . \tag{4.14}$$

In particular, with the two critical priors: (1) the range of value of learnable parameters is tightly constrained by the $\mathcal{L}_2$-regularization of a small weight-decay (*e.g.*, 1×10^{-4} for ImageNet experiments); (2) $|\dot{\gamma}|$ is usually a small value fallen in 10^{-1} level, we suppose $\|\bar{\boldsymbol{x}}\|, \|\boldsymbol{w}\| < 1$ in common, so that $\left|\frac{1+\alpha}{\|\bar{\boldsymbol{x}}\|\|\boldsymbol{w}\|}\right| > |1+\alpha| \geq \frac{1}{4}|\dot{\gamma}|$ (*i.e.*, $\left|\nabla_{\boldsymbol{w}} \bar{s}(\boldsymbol{w})\right| \geq |\nabla_{\boldsymbol{w}} \nu(\boldsymbol{w})|$) can be met easily. This ensures the applicability of the term-*B* to IIEU-B.

Moreover, for $\hat{\varrho}_x < \eta$, the relative relationship of the term-*S* and -*B* about any given $\boldsymbol{w}$ preserves, as they both have

$$\frac{\partial \varsigma}{\partial \hat{\varrho}_x} = \frac{\partial \left(\eta \exp^{\hat{\varrho}_x - \eta}\right)}{\partial (\hat{\varrho}_x - \eta)} \frac{\partial (\hat{\varrho}_x - \eta)}{\partial \hat{\varrho}_x} = \eta \exp^{\hat{\varrho}_x - \eta} , \tag{4.15}$$

which still preserves the applicability of the term-*B* to IIEU-B.

Dynamic Coupler.

Recent neural networks usually leverage the shortcut (*i.e.*, residual) to transmit details of the lower layers to the main branch of the current layer. The estimated II-scores of the features from the main branch and the shortcut, however, are un-calibrated before fusion for their cross-layer information such that they possibly have compromised comparability in terms of importance measure as we propose IIEU mainly to score alternative candidates within the same layer. To address this problem, we propose the **D**ynamic **C**oupler (**DC**) module as

a tailor-made enhancement tool for IIEU-B. DC module is a new lightweight joint-feature-gating model that dynamically rectifies features of the main branch and the shortcut with the channel contexts such that the cross-layer features can be adaptively fused with calibrated intensities. In particular, we refer to the enhanced IIEU-B as IIEU-**DC**.

The DC module works at a low cost, which only employs a joint-channel LayerNorm [128] with a small MLP (with a reduction ratio r defaulted by 16) to project the global channel statistics of the input main and shortcut features to the adaptive channel weights. Specifically, our DC module aims to estimate the channel-wise combination weights dynamically for the effective fusion of the main and the shortcut features by extending the channel attention mechanism [18] from

$$\breve{\boldsymbol{X}}_c = \lambda_{1,c}\tilde{\boldsymbol{X}}_c \oplus \tilde{\boldsymbol{Q}}_c\,, \tag{4.16}$$

i.e., a single-side channel weights estimation without involving the contextual information of the residual feature, to the case

$$\breve{\boldsymbol{X}}_c = \lambda_{1,c}\tilde{\boldsymbol{X}}_c \oplus \lambda_{2,c}\tilde{\boldsymbol{Q}}_c\,, \tag{4.17}$$

i.e., the double-side channel weights estimation that jointly exploits the dual contextual cues of the main branch and the residual features in an interactive manner, where $\oplus$ denotes the element-wise summation. $\tilde{\boldsymbol{X}}_c, \tilde{\boldsymbol{Q}}_c \in \mathbb{R}^{H\times L}$ denote the main branch and residual feature matrices of the c-th channel (*i.e.*, the channel slices of the corresponding feature maps), respectively. $\lambda_{1,c}, \lambda_{2,c} \in \mathbb{R}$ denote the estimated weights for the c-th main branch and the shortcut feature matrices, respectively. $\breve{\boldsymbol{X}}_c \in \mathbb{R}^{H\times L}$ denotes the fused feature matrix of the c-th channel. In particular, we constraint $\lambda_{1,c} + \lambda_{2,c} = 1$ by the Softmax function. Note that besides the clear differences in operations, the motivation of our DC module, *i.e.*, to realize targeted dynamic weighted mixing of the main branch and shortcut features, is also different from the SK-Net [114] which generalizes SE-Net to merge multi-scale features. The diagram of the DC module is depicted in Figure 4.5.

4.4 Related Work

In Section 4.3, we explore the possible working mechanism of neural feature activation from MCDM with supposing $\phi(\tilde{x}) = \varsigma(\hat{\varrho}(\tilde{x}))\,\tilde{x}$. Based on it, we propose to categorize the related methods of activation models by the different adjusters ς or/and approximated ideal similarities $\hat{\varrho}(\tilde{x})$ they introduced. As a prevailing practice, most of the popular methods applied $k\tilde{x}$ as $\hat{\varrho}(\tilde{x})$, where $k \in \mathbb{R}$, and devoted to presenting new variants of ς (*i.e.*, $\phi(\tilde{x}) = \varsigma(k\tilde{x})\,\tilde{x}$). Inspired by neuronal behaviors, ReLU [5] is a maxout approximation to Softplus [38], whose ς is a binary mask of 0 and 1 for $\tilde{x} \leq 0$ and $\tilde{x} > 0$, respectively. LeakyReLU [129] allows slight information leakage from the negative interval to prevent

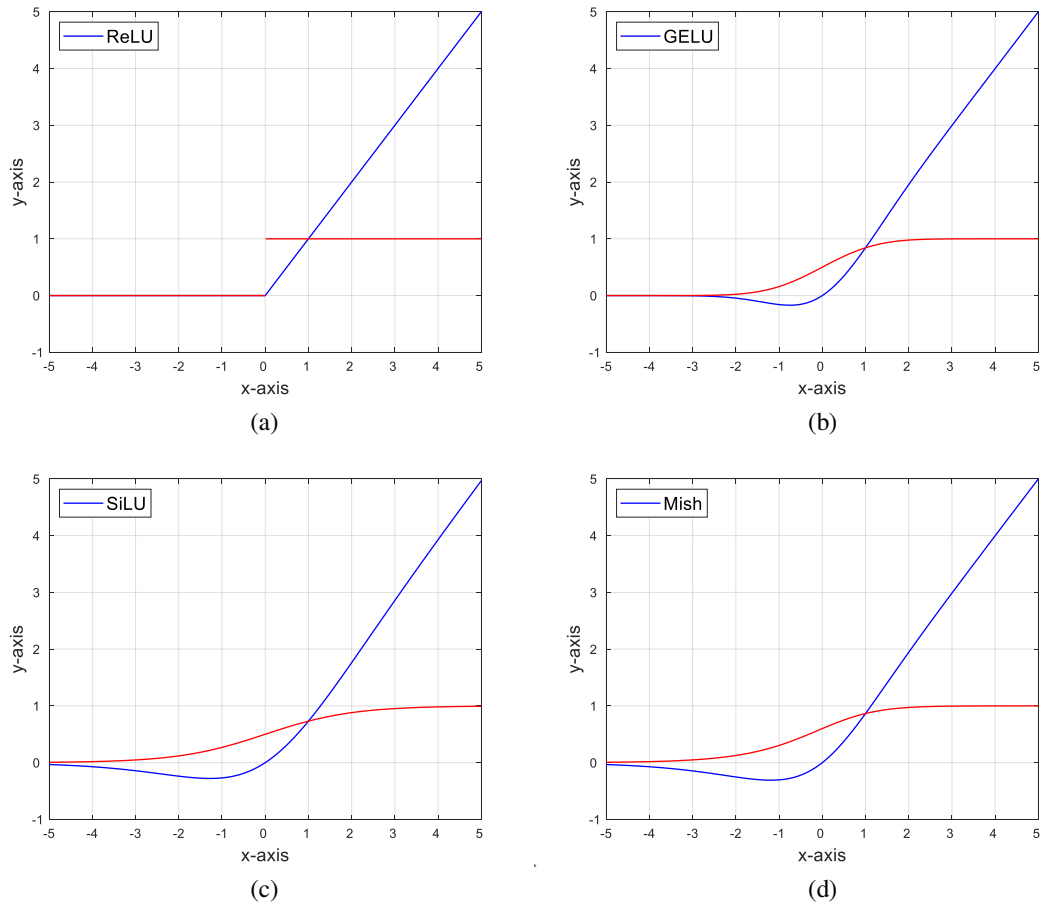


Figure 4.6: Examples of popular activation functions ($\phi(\cdot)$, colored by "**blue**") with their re-weighting functions ($\rho(\cdot)$, colored by "**red**"): (a) ReLU [5]; (b) GELU [12]; (c) SiLU [13]; (d) Mish [14].

dead tensors. PReLU [130] instead learns an adaptive slope for the negative interval. Besides, ELU [131] activates negative $\tilde{x}$ with an exponential function. Goodfellow *et al*. [132] discussed the universal function approximators with piecewise linear components. More recently, PWLU [16] suggested a learnable piecewise linear adjuster. Molina *et al*. [45] proposed a flexible activation function approximator (PAU) based on Padé approximant [133].

ReLU also encouraged recent self-gated activation models. SiLU [13] suggested a Sigmoid ς to enable smooth masking on $\tilde{x}$. Similarly, Swish [6] also considered a Sigmoid-based ς with an updatable slope k for $\tilde{x}$ to enable flexible fitting. Mish [14] proposed a recent smooth ς, *i.e*., $\tanh(\text{softplus}(\cdot))$. ACON-C [7] extended Swish with the learnable upper/lower bounds for the gradient. GELU [12] introduced the first Gauss-Error-Function-based (ERF) smooth ς. GELU also inspired a series of SOTA activation models, *e.g*., ErfAct/Pserf [15] and Smooth Maximum Units (*i.e*., SMU-1 and SMU) [8], which are different kinds of smooth variants/approximations to ReLU and GELU with new ERF-based adjusters. These works achieved clear gains to ReLU networks by introducing flexible smooth adjusters ς. However, as discussed with Intuition 4.3, activation models that apply $k\tilde{x}$ as the approximated similarities $\hat{\varrho}(\tilde{x})$ will encounter the *mismatched feature scoring* problem

which puts an obstacle impeding them from further improvements. Figure 4.6 illustrates 4 popular activation functions ($\phi(\cdot)$) with their re-weighting functions ($\rho(\cdot)$), which includes ReLU [5], GELU [12], SiLU [13], and Mish [14].

Several recent works leveraged attention to activate features, which we treat as presenting a class of approximated similarities $\hat{\varrho}(\tilde{x})$ that tune $\tilde{x}$ with content-based cues. FReLU [41] encoded local spatial cues to rectify $\tilde{x}$ with depth-wise convolutions. DyReLU [42] introduced the SE-Net-based [18] channel attention to improving feature activation. Meta-ACON [7] further extended Swish by generalizing channel attention to learn a dynamic scaling factor for $\tilde{x}$. These works generalized attention to enhance feature activation, provided a promising design space, and realized SOTA gains to ReLU networks. However, as the biasing effects led by the norms occur before the attention, the *mismatched feature scoring* problem remains unsolved. In contrast, IIEU presents the initial solution to the critical problem and achieves the new SOTA improvements with fewer parameters.

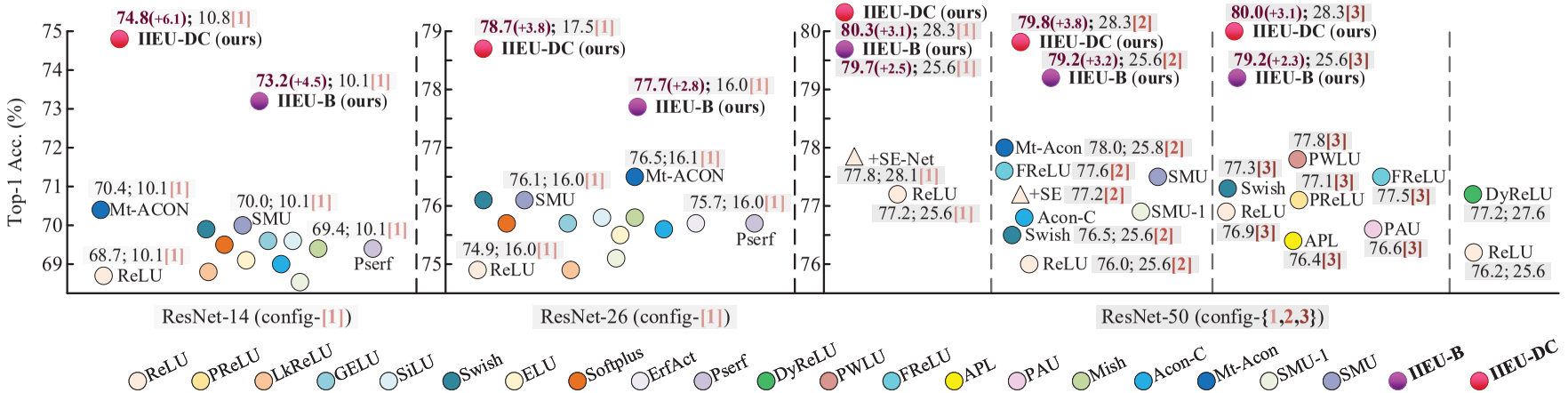
Wu [134] comprehensively analyzed the convergency, stability, and feasibility of the non-monotonic self-gated activation models at a theoretical level, which laid a solid foundation for our exploration. Wu worked to explain past methods while not presenting new activation methods. Concurrently, in a different but related field, Cho *et al*. [135] also found evidence from decision-making to interpret how neural networks capture temporal patterns in channels. We agree with their explanations of neural operations and propose to re-interpret neural feature activation from the new philosophical perspective of MCDM, in which we identify the unexcavated yet critical *mismatched feature scoring* problem and present our new activation model, IIEU, as its solution, enjoying remarkable improvements to the SOTAs, based on our new intuitions and the deduced properties of effective feature activation (*i.e*., selective re-calibration).

4.5 Experiment

We evaluate the effectiveness and versatility of our IIEUs on various vision benchmark datasets: ImageNet [86] and CIFAR-100 [28] (image classification); COCO [23] (object detection); KITTI-Materials [1] (RGB road scene material segmentation). We validate IIEU-B and IIEU-DC through extensive experimental comparisons with the popular and SOTA activation models which include (1) ReLU families: [38, 5, 129, 130]; (2) smooth/self-gated models: [131, 12, 13, 6, 14, 8, 15]; (3) attention-based models: [42, 41, 7]; (4) others: [16, 45, 136]. We validate the core components of our IIEU-B, *i.e*. the proposed adjuster ς and approximated similarity measure $\hat{\varrho}_x$ through targeted ablation studies.

4.5.1 ImageNet Classification

Implementation details. We evaluate our IIEUs with three kinds of networks of different sizes, *i.e*., the popular ResNet [11] and the lightweight MobileNetV2 [9] and ShuffleNetv2 [10], where the baselines use ReLU [5] for activation. To ensure fair comparisons



		ReLU	PReLU	LkReLU	GELU	SiLU	Swish	ELU	Softplus	ErfAct	Pserf	DyReLU	PWLU	FReLU	APL	PAU	Mish	Acon-C	Mt-Acon	SMU-1	SMU	**IIEU-B**	**IIEU-DC**
RN14	Top-1(%)	68.7	NaN	68.8	69.6	69.6	69.9	69.1	69.5	NaN	69.4	N/A	N/A	N/A	N/A	N/A	69.4	69.0	70.4	68.5	70.0	**73.2**(+4.5)	**74.8**(+6.1)
	Param(M)	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	N/A	N/A	N/A	N/A	N/A	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.1[1]	10.8[1]
RN26	Top-1(%)	74.9	NaN	74.9	75.7	75.8	76.1	75.5	75.7	75.7	75.7	N/A	N/A	N/A	N/A	N/A	75.8	75.6	76.5	75.1	76.1	**77.7**(+2.8)	**78.7**(+3.8)
	Param(M)	16.0[1]	16.0[1]	16.0[1]	16.0[1]	16.0[1]	16.0[1]	16.0[1]	16.0[1]	16.0[1]	16.0[1]	N/A	N/A	N/A	N/A	N/A	16.0[1]	16.0[1]	16.1[1]	16.0[1]	16.0[1]	16.0[1]	17.5[1]
RN50	Top-1(%)	77.2	77.1	N/A	N/A	N/A	77.3	N/A	N/A	N/A	N/A	77.2	77.8	77.6	76.4	76.6	N/A	76.8	78.0	76.9	77.5	**79.7**(+2.5)	**80.3**(+3.1)
	Param(M)	25.6[1]	25.6[3]	N/A	N/A	N/A	25.6[3]	N/A	N/A	N/A	N/A	27.6	N/A[3]	25.7[2]	25.6[3]	25.6[3]	N/A	25.6[2]	25.8[2]	25.6[2]	25.6[2]	25.6[1]	28.3[1]

Figure 4.7: Comparison of different activation models with **R**es**N**et (**RN**) backbones on ImageNet. **IIEU-B** and **-DC** are ours; ErfAct/Pserf (AAAI'22) [15], ACON-C/Mt-ACON (*i.e.*, Meta-ACON, CVPR'21) [7], PWLU (ICCV'21) [16], and SMU-1/SMU (CVPR'22) [8] are SOTAs. We train our and compared activation models which have the public official projects with RN-14 and -26 from scratch using ***cfg-1*** [17] and report the results by "Top-1 Acc.(%); Params.(M)[cfg]", ***where "(+·)" show the improvements in Top-1 Acc. of our IIEUs over the ReLU baselines***. For RN-50, we report the official results for all the compared models (including the ReLU baselines w/ or w/o SE-Net [18]) and implemented results for IIEUs with ***cfg-1*** [17], ***-2*** [7], and ***-3*** [16], respectively. "NaN" denotes failed training; "N/A" means non-applicable/unknown. ©2023 IEEE [2]

Table 4.1: Comparison of different activation models on ImageNet using lightweight backbones. We train each of the networks with our IIEUs and popular/SOTA act models from scratch using cfg-*c*. For SOTA competitors (Pserf (AAAI'22) [15] and SMU-1/SMU (CVPR'22) [8]), we adopt their official model settings (*i.e.*, the initialization strategies for learnable parameters and values of the hyper-parameters). ©2023 IEEE [2]

Activation	MobileNetV2 0.17× [9]		ShuffleNetV2 0.5× [10]	
	#Params.	Top-1(%)↑	#Params.	Top-1(%)↑
ReLU [5]	1.4M	49.7	1.4M	59.9
GELU [12]	1.4M	52.9	1.4M	61.5
Swish [6]	1.4M	53.7	1.4M	61.8
Mish [14]	1.4M	53.1	1.4M	61.5
Pserf [15]	1.4M	52.6	1.4M	60.8
SMU-1 [8]	1.4M	51.7	1.4M	60.2
SMU [8]	1.4M	54.2	1.4M	61.8
IIEU-B (Ours)	1.5M	**58.0**(+8.3)	1.4M	**65.8**(+5.9)
IIEU-DC (Ours)	1.5M	**58.1**(+8.4)	1.4M	**66.8**(+6.9)

with existing activation models trained with various configures, we adopt three different basic configures applied in [17], [7], and [16] (**denoted by *cfg*-1, -2, and -3**, respectively) to train ResNets equipped with IIEU-B and IIEU-DC, respectively. The details of implementation configures are described as follows.

1. **cfg-1** applies 120 epochs using the basic SGD optimizer with the weight decay of 1^{-4} and momentum of 0.9, where the first 5 epochs are the linear warm-up epochs. The learning rate starts from 0.1 with a batch size of 256 by default and decays to 1^{-5} following the cosine schedule. After the main training schedule, it applies an extra 10 cool-down epochs with the minimum learning rate 1^{-5} to stabilize the model weights. It follows the common practice to first randomly resize input images and then crop them to a size of 224×224. In the test phase, input images are center cropped to 224×224. It adopts the standard data augmentation strategy used in [114, 17, 7, 18].
2. **cfg-2** has two differences compared to **cfg-1**: (1) it applies the linear learning rate schedule which starts from 0.1 and decays to 1^{-5} (*i.e.*, the minimum learning rate); (2) it removes the extra 10 cool-down epochs.
3. **cfg-3** has one difference compared to **cfg-1**: (1) it applies a cosine learning rate with only 100 epochs.

We train MobileNetV2(s) and ShuffleNetV2(s) with two different configures, where the former is a standard configure used in [137, 9, 10, 8, 41, 7] and the later replaces the linear learning rate scheduler in the former with the cosine learning rate scheduler (denoted by **cfg**-*l* and –*c*, respectively). We detail the **cfg**-*l* and **cfg**-*c* as follows:

1. **cfg**-*l* applies the basic SGD optimizer with the weight decay of 4×10^{-5} and momentum of 0.9. Each network is trained with a batch size of 1024 for 300k iterations (*i.e.*, 240

Table 4.2: Comparison of activation models with ***cfg-2*** [7]. ***We compare IIEUs with ResNet-26 and -50 backbones*** to the official results of the popular/SOTA activation models with the large ResNet-101. ©2023 IEEE [2]

Activation	Backbone	#Params.	Top-1(%)↑
ReLU [5]	ResNet-101 [11]	44.5M	77.2
PReLU [130]		44.5M	77.3
Swish [6]		44.5M	77.3
FReLU [41]		45.0M	77.9
ACON-C [7]		44.6M	77.9
Meta-ACON [7]		44.9M	78.9
IIEU-B (Ours)	**ResNet-50** [11]	**25.6**M	**79.2**
IIEU-DC (Ours)		**28.3**M	**79.8**
IIEU-B (Ours)	**ResNet-26** [11]	**16.0**M	**77.3**
IIEU-DC (Ours)		**17.5**M	**78.3**

Table 4.3: Comparing IIEUs with ReLU baseline and SOTA activation models on ShuffleNetV2 [10] with ***cfg-l*** [9]. ©2023 IEEE [2]

Activation	Mish[14]	SMU-1[8]	SMU[8]	Mt-AN[7]	ReLU[5]	**IIEU-B**	**IIEU-DC**
Backbone	ShuffleNetV2 1.0× [10]					ShuffleNetV2 1.0× [10]	
#Params.	2.3M	2.3M	2.3M	2.6M	2.3M	2.5M	2.6M
Top-1(%)↑	70.5	71.2	71.9	72.1	69.4	**73.3**(+3.9)	**74.0**(+4.6)

epochs as for the number of images in the training set of ImageNet). The learning rate starts from 0.5 and decreases to 1^{-5} by following the linear schedule. It follows the common practice to first randomly resize the input images and then crop them to a size of 224×224. In the test phase, input images are center cropped to 224×224. It adopts the standard data augmentation strategy used in [114, 17, 7, 18].

2. **cfg**-*c* is obtained by replacing the linear LR scheduler with the cosine LR scheduler.

Main results. Figure 4.7 and Tab. 4.1, 4.2, 4.4, 4.3 report the comparative results of our and the popular/SOTA activation models with various networks on ImageNet, where we have three major observations: (1) IIEUs remarkably improve the popular and SOTA activation models on different networks yet add negligible/marginal parameters and FLOPs to the ReLU baselines which represent the relatively lowest computations (as shown in Table 4.5). On ResNet-14, MobileNetV2 $0.17\times$, and ShuffleNetv2 $0.5\times$, IIEU-B and -DC improve ReLU by $\{\mathbf{4.5\%}, \mathbf{8.3\%}, \mathbf{5.9\%}\}$ and $\{\mathbf{6.1\%}, \mathbf{8.4\%}, \mathbf{6.9\%}\}$, respectively. It is worth noting that the improvements of our IIEUs to the SOTAs on some of the networks (*e.g.*, MobileNetV2, ResNet-14, and ResNet-26) are more significant than the SOTAs to the ReLU baselines. (2) With IIEUs, the small ResNet-26s outperform/match the deeper ResNet-50s and -101s with the SOTA activation models and the ResNet-50s enjoy clear improvements over the large ResNet-101s, where IIEU-B and -DC achieve the high Top-1

Table 4.4: Comparing IIEUs with ReLU baseline and SOTA activation models on MobileNetV2 (MNV2) with ***cfg-l*** [9]. ©2023 IEEE [2]

Activation	MobileNetV2 0.17× [9]		MobileNetV2 1.0× [9]	
	#Params.	Top-1(%)↑	#Params.	Top-1(%)↑
PWLU [16]	N/A	N/A	N/A	74.7
ACON-C [7]	1.5M	51.1	3.6M	73.6
Meta-ACON [7]	1.9M	53.8	3.9M	75.0
ReLU [5]	1.4M	49.7	3.5M	72.1
IIEU-B (Ours)	1.5M	**58.1**(+8.4)	3.6M	**75.8**(+3.7)
IIEU-DC (Ours)	1.5M	**58.7**(+9.0)	3.6M	**76.2**(+4.1)

Table 4.5: Comparisons of FLOPs and parameters of IIEUs with ReLU on ResNet backbones. We show the official Top-1 of the ReLU ResNet-50 adopted from [17]. All the models are trained by the **cfg-1** [17] (including the ReLU ResNet-50). ©2023 IEEE [2]

Activation	Metric	ResNet-14 [11]	ResNet-26 [11]	ResNet-50 [11]
ReLU [5]	#Params.	10.1M	16.0M	25.6M
IIEU-B		10.1M	16.0M	25.6M
IIEU-DC		10.8M	17.5M	28.3M
ReLU [5]	FLOPs	1.5G	2.4G	4.1G
IIEU-B		1.5G	2.4G	4.2G
IIEU-DC		1.5G	2.4G	4.2G
ReLU [5]	Top-1(%)↑	68.7	74.9	77.2
IIEU-B		**73.2**	**77.7**	**79.7**
IIEU-DC		**74.8**	**78.7**	**80.3**

Acc. of {$\mathbf{79.7}\%$, $\mathbf{79.2}\%$, $\mathbf{79.2}\%$} and {$\mathbf{80.3}\%$, $\mathbf{79.8}\%$, $\mathbf{80.0}\%$} trained with cfg-1, -2, and -3, respectively. (3) IIEUs are highly stable with different training configures and consistently outperform the SOTA activation models by a clear margin on different networks. In the subsequent text, we present a detailed **convergence analysis**, where our IIEUs not only reach the highest Top-1 Acc. but also draw the steepest slopes of optimization. This validates the effectiveness of our IIEU for neural feature activation.

Convergence analysis. Figure 4.8 depicts the convergence trends in Top-1 accuracy (the higher the better) and training loss (the lower the better) of ResNet-14 and ResNet-26 backbones equipped with our IIEUs and the compared activation models, respectively. Each model is trained by the **cfg-1** [17] from scratch to convergence, respectively. ReLU networks are the baselines and Pserf (AAAI'22) [15], ACON-C/Mt-ACON (*i.e.*, Meta-ACON, CVPR'21) [7], and SMU-1/SMU (CVPR'22) [8] are other SOTAs. It is worth noting that our IIEU-B and IIEU-DC consistently achieve the comparatively highest Top-1 accuracies and lowest loss values over the varying epochs.

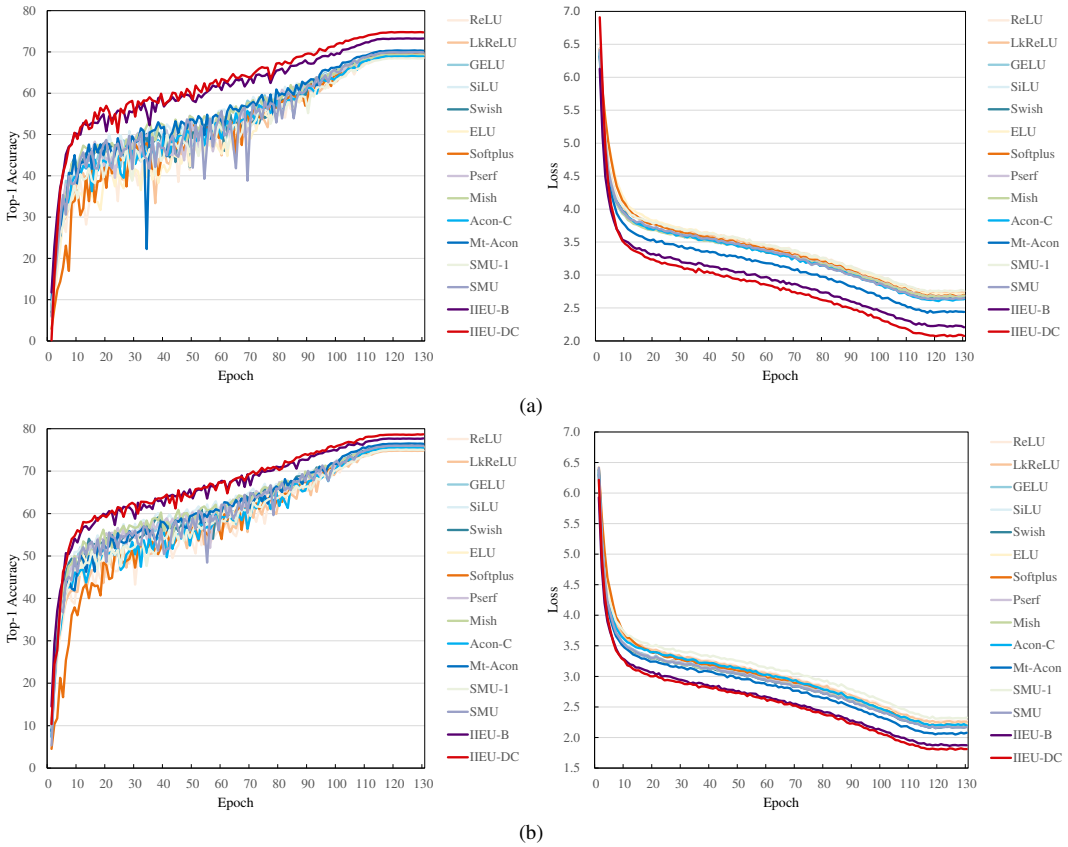


Figure 4.8: Top (a): the accuracy curve (left) and loss curve (right) of ResNet-14 backbone with different activation models. Bottom (b): the accuracy curve (left) and loss curve (right) of ResNet-26 backbone with different activation models. ©2023 IEEE [2]

Tables 4.6 and 4.7 reports the number of training epochs to convergence for the networks (*i.e.*, ResNet-14 and ResNet-26, respectively) of different activation models, where we select the epoch that *each corresponding network reaches its lowest training loss value* as the criterion of *convergence*. Moreover, for detailed comparisons of convergence speed, we also show the specific epochs that the loss of each network first drops below the specific values (*i.e.*, $\text{epoch}_{\mathcal{L}<3.0}$, $\text{epoch}_{\mathcal{L}<2.5}$, and $\text{epoch}_{\mathcal{L}<2.0}$ are selected, where $\mathcal{L}$ denotes "loss value"). Our two major observations are: (1) IIEU-B and IIEU-DC demonstrate improved convergence properties. That is, IIEU-B and IIEU-DC reach each of the corresponding loss thresholds with relatively fewer training epochs. (2) IIEU-B and IIEU-DC reach clearly lower minimum training loss values (*i.e.*, $\mathcal{L}_{min}$) than other compared SOTA/popular/baseline activation models. This validates the convergence property of IIEU.

4.5.2 CIFAR-100 Classification

Implementation details. We evaluate different activation models with the public CIFAR versions [29] of ResNets and ShuffleNetV2, which contain fewer parameters than the ImageNet networks. For fair comparisons, we adopt the standard data augmentations used

Table 4.6: Convergence analysis of different activation models with ResNet-14 backbone. $\mathcal{L}$ denotes "loss value." We show the minimum values of training loss $\mathcal{L}_{min}$ reached by different activation models with two decimal places. "–" denotes "unreachable." Note that each model is trained for 130 epochs using **cfg-1** [17]. ©2023 IEEE [2]

Metric	ReLU	GELU	Swish	Pserf	Mish	Mt-ACON	SMU	**IIEU-B**	**IIEU-DC**
$\mathcal{L}_{min} \downarrow$	2.74	2.66	2.65	2.66	2.66	2.43	2.63	**2.21**	**2.07**
$\text{epoch}_{\mathcal{L}_{\min}}$	119	122	117	117	122	117	119	130	124
$\text{epoch}_{\mathcal{L}_{<3.0}} \downarrow$	98	93	91	93	93	79	91	**57**	**44**
$\text{epoch}_{\mathcal{L}_{<2.5}} \downarrow$	–	–	–	–	–	112	–	**97**	**90**
$\text{epoch}_{\mathcal{L}_{<2.0}} \downarrow$	–	–	–	–	–	–	–	–	–
Top-1(%)↑	68.7	69.6	69.9	69.4	69.4	70.4	70.0	**73.2**	**74.8**

Table 4.7: Convergence analysis of different activation models with ResNet-26 backbone. $\mathcal{L}$ denotes "loss value." We show the minimum values of training loss $\mathcal{L}_{min}$ reached by different activation models with two decimal places. "–" denotes "unreachable." Note that each model is trained for 130 epochs using **cfg-1** [17]. ©2023 IEEE [2]

Metric	ReLU	GELU	Swish	Pserf	Mish	Mt-ACON	SMU	**IIEU-B**	**IIEU-DC**
$\mathcal{L}_{min} \downarrow$	2.26	2.18	2.15	2.19	2.17	2.05	2.15	**1.86**	**1.80**
$\text{epoch}_{\mathcal{L}_{\min}}$	119	119	119	119	120	126	119	124	126
$\text{epoch}_{\mathcal{L}_{<3.0}} \downarrow$	68	57	53	57	56	46	53	**23**	**20**
$\text{epoch}_{\mathcal{L}_{<2.5}} \downarrow$	102	96	96	98	96	90	95	**74**	**71**
$\text{epoch}_{\mathcal{L}_{<2.0}} \downarrow$	–	–	–	–	–	–	–	**107**	**104**
Top-1(%)↑	74.9	75.7	76.1	75.7	75.8	76.5	76.1	**77.7**	**78.7**

in [114] to train all the networks with our and compared activation models by a basic SGD optimizer with the weight decay of 5×10^{-4} and momentum of 0.9. Each model is trained for 350 epochs with a batch size of 256. The learning rate starts from 0.1 and decreases to 1^{-6} following the cosine schedule. All the input images are fixed to the size of 32×32.

Experimental results. As shown in Table 4.8, our IIEUs significantly improve all the popular and SOTA activation models on various networks. These experimental results are highly consistent with the ImageNet evaluations. It is worth noting that IIEUs show superior stability to the compared SOTA self-gated and attention-based activation models, as IIEUs demonstrate higher consistency of the improvements on the corresponding ImageNet and CIFAR networks. This validates the scalability of IIEUs for datasets of different sizes.

4.5.3 Ablation Study

Approximated similarity $\hat{\varrho}_x$. $\hat{\varrho}_x$ serves as the core of the II-score estimation for IIEU. Here we discuss $\hat{\varrho}_x$ with three targeted control groups using CIFAR-ResNet-56: (1) $\phi(\tilde{x}) = \varsigma(\tilde{x})$ (denoted by **Act-**ς), *i.e.*, let the adjuster ς apply individually without the proposed $\hat{\varrho}_x$ such that IIEU-B degrades to a simpler parametric activation model; (2) replacing ς by ReLU (denoted by "(-R)"); (3) Without ς. As shown in Table 4.9, Act-ς shows a significant

Table 4.8: Comparison of different activation models on CIFAR-100. We train each model 8 times and report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1. ©2023 IEEE [2]

Activation	CIFAR-ResNet-29 [11]		CIFAR-ResNet-56 [11]	
	#params.	Top-1(%)↑	#params.	Top-1(%)↑
ReLU [5]	0.3M	70.5 ± 0.3	0.6M	74.4 ± 0.3
ELU [131]	0.3M	72.6 ± 0.2	0.6M	74.7 ± 0.3
PReLU [130]	0.3M	70.1 ± 0.5	0.6M	73.2 ± 0.4
GELU [12]	0.3M	71.4 ± 0.3	0.6M	75.3 ± 0.3
SiLU [13]	0.3M	72.0 ± 0.4	0.6M	75.3 ± 0.4
Swish [6]	0.3M	71.5 ± 0.3	0.6M	74.8 ± 0.2
Mish [14]	0.3M	72.1 ± 0.3	0.6M	75.2 ± 0.3
SMU [8]	0.3M	71.1 ± 0.4	0.6M	74.9 ± 0.3
SMU-1 [8]	0.3M	70.7 ± 0.3	0.6M	74.7 ± 0.2
Pserf [8]	0.3M	71.6 ± 0.2	0.6M	75.3 ± 0.2
ACON-C [7]	0.3M	70.9 ± 0.2	0.6M	74.1 ± 0.3
Meta-ACON [7]	0.3M	72.2 ± 0.3	0.6M	75.7 ± 0.2
IIEU-B (Ours)	0.3M	$\mathbf{74.7 \pm 0.3}$	0.6M	$\mathbf{77.2 \pm 0.3}$
IIEU-DC(Ours)	0.3M	$\mathbf{75.8 \pm 0.4}$	0.6M	$\mathbf{78.1 \pm 0.2}$

Table 4.9: Ablation study on $\hat{\varrho}_x$ and ς. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1 accuracy for each model. ©2023 IEEE [2]

Prototype	$\varsigma(\tilde{x})$		$\varsigma(\hat{\varrho}_x)$			$\hat{\varrho}_x$
ς	ReLU	Raw ς	Sigmoid	ReLU	Raw ς	Identity
#Params.	0.6M	0.6M	0.6M	0.6M	0.6M	0.6M
Top-1(%)↑	74.4±0.3	73.2±0.3	74.5±0.2	77.0±0.4	$\mathbf{77.2 \pm 0.3}$	76.6±0.2

drop in accuracy compared to the original IIEU-B. In contrast, model (-R) that preserves the approximated similarity $\hat{\varrho}_x$ shows slight accuracy decreases. Without ς, the control group (3) still improves the ReLU baseline by a large margin. The experimental results are consistent with our hypothesis, where $\hat{\varrho}_x$ of IIEU is supposed to introduce the main accuracy gains and the ς serves as a helper function to ensure Property 4.1 which is possibly met conditionally without ς.

Adjuster ς. We further discuss ς by replacing it with the Sigmoid function (denoted by δ, *i.e.* the smooth adjuster ς of SiLU [13] and Swish [6]). Table 4.9 reports the comparative results of different control groups, where our original ς outperforms Sigmoid function by a large margin. It is worth noting that our $\hat{\varrho}_x$ also achieves competitive Top-1 with ReLU-based ς. These results are in line with Intuition 4.6, as our ς and ReLU function are both conditionally linear about $\hat{\varrho}_x$ for $\hat{\varrho}_x \leq 0$, while the slope of Sigmoid gradually declines with the increases of $\hat{\varrho}_x$ if $\hat{\varrho}_x > 0.5$. Moreover, in contrast to ReLU, our original ς introduces further improvements with the adaptive threshold η.

Table 4.10: Ablation study on the term-*S* and term-*B*, where we report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1 accuracy for each model. ©2023 IEEE [2]

IIEU-B	Term-*S*		Positive-constraint on Term-*B*		
	W/ (Raw)	***W/o***	(a) δ (Raw)	(b) Softplus	(c) W/o
#Params.	0.6M	0.6M	0.6M	0.6M	0.6M
Top-1(%)↑	$\mathbf{77.2 \pm 0.3}$	$\mathbf{32.6 \pm 0.4}$	$\mathbf{77.2 \pm 0.3}$	76.8 ± 0.3	75.8 ± 0.2

W/ or W/o term-*S*. We suppose that the term-*S* (Equation (4.3)) which serves as the main term of the II-score estimation introduces the main accuracy gains. We validate term-*S* by comparing IIEU-B to the abridged IIEU-B which removes the term-*S*. As shown in Table 4.10, **removing term-*S* will cause a dramatic drop in accuracy**, which is consistent with our intuition.

Positive constraint on term-*B*. We suppose a bounded and positive term-*B* is helpful for the II-score estimation and choose Sigmoid function to cast effective positive constraint (pos-cst) on term-*B* (Section 4.3.2). Herein, we further investigate the selection for positive constraint by replacing Sigmoid (denoted by (a)) with two tailored control groups: (b) Softplus function; (c) identity (*i.e.*, without positive constraint). Experimental results in Table 4.10 demonstrate that Acc. (a) $>$ Acc. (b) $>$ Acc. (c), which is in line with our intuition, as we suppose the term-*B* needs to be less influential on filter updating than the term-*S* , which contributes to preventing the harmful neutralization effect. That is, (1) as term-*B* with (b) can have significantly higher gradients than (a), we suppose it to show inferior accuracy to (a); (2) we expect (c) to show comparatively worst accuracy, as it likely violates the Intuition 4.6 with outputting negative values.

Table 4.11: Ablation study on normalization operations of the term-*B* in IIEU-B. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1. ©2023 IEEE [2]

(1) LN	(2) GN			(3) BN	(4) Blank	(5) ReLU
	$G = C$	$G = 4$	$G = 2$			
0.6M	0.6M	0.6M	0.6M	0.6M	0.6M	0.6M
$\mathbf{77.2 \pm 0.3}$	76.4 ± 0.2	76.9 ± 0.2	77.0 ± 0.3	75.4 ± 0.3	76.6 ± 0.3	74.4 ± 0.3

Alternative normalization operations of term-*B*. We consider Layer Normalization (LayerNorm) [128] as an effective operation for the term-*B* (*i.e.*, ν) of IIEU-B to perform flexible channel-dependent scaling and shift to channel statistics with negligible cost (introduced in Formulation, Section 2.3). Herein, we further investigate the effectiveness (*i.e.*, suitability) of LayerNorm for the learning of term-*B* by comparing it to different relevant parametric normalization operations that are commonly applied in neural networks. Specifically, a targeted ablation study of applying alternative parametric normalization layers in term-*B* of

Table 4.12: Comparison of different activation models on the COCO object detection [23] using RetinaNet [24] with ResNet-50 [11] backbone. ©2023 IEEE [2]

Activation	#Params.	mAP (%)↑	AP_{50}(%)↑	AP_{75}(%)↑	AP_S(%)↑	AP_M(%)↑	AP_L(%)↑
ReLU [5]	37.7M	36.7	56.0	39.3	21.0	40.2	48.2
Swish [6]	37.7M	37.2	56.3	39.9	21.0	41.1	47.8
SMU [8]	37.7M	37.5	56.6	40.2	21.5	41.5	48.4
Mt-ACON [7]	37.9M	36.5	55.9	38.9	19.9	40.7	50.6
IIEU-B	37.7M	**38.2**	**58.2**	**40.6**	**23.2**	**42.1**	**49.2**
IIEU-DC	40.4M	**38.6**	**59.0**	**40.8**	**22.2**	**42.6**	**50.7**

IIEU-B is conducted on CIFAR-100 [28] dataset with CIFAR-ResNet-56 backbone [11, 29], where five **c**ontrol **g**roups (**cg**) are set up: **(1) LayerNorm [128] (*i.e.*, the original setting);** (2) Group Normalization (GroupNorm) [138] with groups (denoted by G) 2, 4, and C; (3) Batch Normalization (BatchNorm) [127]; (4) the blank group which applies updatable element-wise affine but removing the normalization operation (*i.e.*, Z-Scoring); (5) the ReLU [5] baseline.

We report $\mathbf{mean} \pm \mathbf{std}$ of the Top-1 accuracy in Table 4.11, where our five major observations are: **(1) the LayerNorm group (cg-1) achieves the highest Top-1 accuracy of all the compared groups;** (2) the GroupNorm group (**cg-2**) demonstrates inferior Top-1 accuracy with $G = C$ while yields close accuracies with $G = 2$ and $G = 4$; (3) the BatchNorm group (**cg-3**) shows relatively low Top-1 accuracy; (4) the blank group (**cg-4**) improves the ReLU baseline (**cg-5**) by a large margin and also clearly outperforms the BatchNorm group; (5) **cg-1** to **cg-4** all enjoy clear accuracy improvements to the ReLU baseline. Note that for single vector input (*i.e.*, the case of the term-*B* in IIEU-B), (1) "GroupNorm of $G = 1$" equals to "LayerNorm;" (2) "GroupNorm of $G = C$" equals to "using biases only;" (3) Instance Normalization (InstNorm) is non-applicable. This validates LayerNorm for the learning of the adaptive shift (*i.e.* term-*B*) in IIEU-B.

4.5.4 MS COCO Object Detection

Implementation details. As generic activation models, our IIEUs can be easily extended to other vision tasks. We evaluate our IIEU-B and IIEU-DC on MS COCO [23] object detection using the popular efficient detector RetinaNett [24]. We compare our IIEUs to the baseline ReLU [5], the popular Swish [6], and the current SOTAs Meta-ACON [7] and SMU [8]. For fair comparisons, we adopt the default implementation configures ($1\times$ schedule) defined by the MMDetection toolbox [139] and report the standard evaluation metrics, *i.e.*, mAP (the primary metric of averaged precisions), AP_{50}, AP_{75}, AP_S, AP_M, AP_L (specific APs at different scales). We employ the ResNet-50 [11] backbones equipped with different activation functions, each applied with their corresponding ImageNet pre-trained weights. Note that we keep using the deterministic mode for each of the implementations to ensure reproducibility.

Experimental results. We show the experimental results in Table 4.12, where our IIEUs enjoy clear gains in accuracy compared to different baseline/popular/SOTA activation models. This validates the scalability and versatility of IIEU. Note that we report the official results for Meta-ACON (*i.e.*, Mt-Acon) as our re-implemented results are lower (possibly caused by the different implementation environments).

4.5.5 KITTI-Materials Road Scene Material Segmentation

Implementation details. We evaluate our and compared activation models on an emerging task, *i.e.*KITTI-Materials [1] RGB road scene material segmentation, using ResNet-50 [11] as the encoder and the efficient multi-level ALL-MLP decoder [4]. For fair comparisons, we adopt the official implementation protocols [1].

Table 4.13: Comparison of different activation models on KITTI-Materials [1] RGB road scene material segmentation. ©2023 IEEE [2]

Activation	Encoder	Decoder	#Params.	mIoU(%)↑
ReLU [5]	ResNet-50 [11]	ALL-MLP [4]	31.7M	40.2
Swish [6]			31.7M	41.2
SMU [8]			31.7M	40.6
Meta-ACON [7]			31.9M	41.7
IIEU-B (Ours)			31.7M	**42.4**

Experimental results. We report the results of our IIEU-B and the compared activation models in Section 4.5.5, where IIEU-B achieves significant accuracy gains to ReLU baseline and also shows clear improvements on SOTAs Swish, SMU, and Meta-ACON. It is worth noting that our IIEU shows consistent significant accuracy improvements on various vision benchmark datasets to the baselines and SOTA activation models.

4.6 Summary

In this Chapter, we propose to interpret neural feature activation from the new perspective of multi-criteria decision-making, where we identify the critical yet unsettled problem, *i.e.* *mismatched feature scoring*, and present our activation model IIEU to solve it with new intuitions and their induced corresponding properties of effective neural feature activation. We validate our practical IIEUs and hypotheses through comprehensive experimental analysis and extensive comparisons with popular and SOTA activation models on various vision benchmark datasets, where IIEUs achieve the new SOTA improvements to the ReLU baseline and also significantly outperform the current popular and SOTA activation models.

Chapter 5

Learning Discriminative Neural Activation With an Adaptive Shift Factor

5.1 Background

Nonlinear **A**ctivation (**Act**) models (functions) are indispensable for the learning of discriminative neural features [30, 45, 33, 46, 32, 31, 32, 132]. Neuronal behaviors [39, 40] originate traditional activation models, *e.g.*, Softplus [38] and ReLU [5], which are fixed and monotonic in calculations. To realize finer rectifications, recent works investigated self-gated-style activation functions based on the general prototype

$$\phi\left(x\right) = \varsigma\left(x\right) x\,, \tag{5.1}$$

where $x \in \mathbb{R}$ is a given feature unit (*i.e.*, scalar), $\phi : \mathbb{R} \to \mathbb{R}$ denotes the applied activation function of x, and $\varsigma : \mathbb{R} \to \mathbb{R}$ defines the re-weighting function of ϕ. As a special case, ReLU can be included in this prototype by specifying $\varsigma\left(x\right)$ as a binary masking of 0 and 1 for $x \leq 0$ and $x > 0$, respectively. Despite the broad applicability, ReLU leaves two practical constraints on neural activation from (1) its rigid masking on positive features, *i.e.*, unified weight assignments that possibly neutralize the discriminativeness, and (2) hard-zero-truncation on negative features that possibly leads to the "dead tensors" problem.

Recent methods addressed these by introducing smooth re-weighting functions with two assumed properties:

1. $\varsigma(x)$ is bounded (typically, $\varsigma(x) \in (0, 1)$);
2. $\varsigma(x)$ is monotonically non-decreasing about x.

These properties theoretically ensure the stability and convergence of neural activation in training [134] and identify typical self-gated activation functions (*e.g.*, [14, 13, 12]) that favor feature rectifications by leaving more flexibility. However, typical self-gated functions can still fall short in adaptability to highly variational training conditions due to fixed re-weighting processes. SOTA methods [15, 7, 8] studied leveraging attention and updatable scaling/bias to enhance self-gated re-weighting by infusing more flexible inductive biases. Although effective, the substantial improvements are still hindered by the critical challenge of norm-induced *mismatched feature scoring* (introduced in Intuition 4.3) invisible to pure biological intuitions. We identified the above problem based on the proposed MCDM hypothesis (introduced in [2], *i.e.*, Chapter 4), where activation models were regarded as selective re-calibrators that emphasize and suppress features based on their importance scores measured by the feature-filter similarities. With this new perspective, we found that differentiated feature and filter norms possibly bias the similarities modeled with feature-filter inner products significantly, hence taking away from how important the features actually are. This inspired a rectified self-gated prototype of activation, *i.e.*,

$$\phi(x) = \varsigma(\varrho(x))\, x\,, \tag{5.2}$$

where $\varrho(x)$ is assumed as an unknown unbiased (*i.e.*, ideal) similarity measure of x and ς preserves the property of monotonically non-decreasing of $\varsigma(x) = \varsigma(\varrho(x))$ to $\varrho(x)$, instead of x the biased similarity. In particular, by designating $\varrho(x) = x$, prototype 5.2 regresses to the base form 5.1. The corresponding method, IIEU ([2], *i.e.*, Chapter 4), addressed the *mismatched feature scoring* problem by approximating $\varrho(x)$ with an adaptive **norm-decoupled importance measure** adjusted with non-local cues, thus performing enhanced feature re-calibrations with the rectified importance scores. Although effective, the brute-force-style norm-decoupling in IIEU inevitably leads to extra runtime for training, due to the relatively complex gradient led by the norm-decoupled approximated similarity $\varrho(x)$.

In this Chapter, we present a novel activation prototype, namely, AdaShift (defined by Equation (5.6)), to address the critical *mismatched feature scoring* problem in a simple yet effective manner with new intuitions in line with the MCDM hypothesis Chapter 4. Specifically,

(1) we suppose prototype 5.1 with properties 1 and 2 imply a critical condition that for an activation process, we have "the larger x, the more important x is," as a re-weighting function ς monotonically re-calibrates x according to its intensity.

(2) By following MCDM hypothesis, we suppose that the importance measure of x is possibly inconsistent with the intensity of x, as the feature/filter norms influenced by the learning states of past layers and initializations can bias the current feature-filter similarity. Yet, unlike IIEU (Chapter 4), here we argue that feature/filter norms provide informative cues for discriminative activations and explicitly decoupling norms can constrain neural features in representational capability. We identify this by rethinking the relationships of feature and filter norms from a common Softmax-based classification in a network, where we find *feature and filter norms present local and non-local cues for classifying output features, respectively*, and by generalize this understanding to general leaning blocks.

Based on the above assumptions (1) and (2), in AdaShift, we introduce an adaptive shift factor Δ, leveraged on the complementary tensor-level non-local context, which learns to approximate $\varrho(x)$ by $\hat{\varrho}(x) = x + \Delta$, thus imposing dynamic inductive biases to a monotonic curve ς to rectify its intensity-based re-weighting on x by exploiting different ranges of local/non-local context of the current learning states in an interactive manner. We identify that Δ can be effectively learned to introduce remarkable improvements to networks by even surprisingly simple approaches that aggregate tensor-level channel/spatial interactions, *e.g.*, only by a vanilla LayerNorm [128] operator applied on a vector of channel statistics (*e.g.*, channel mean responses), with negligible parameters and computational cost. This allows us to propose a brand-new class of activation models, *i.e.*, practical **AdaShift**(s) by embodying the shift factor Δ with different derivatives. In particular, we mainly present two practical AdaShifts as examples, where we refer to the one that solely casts an embedded LayerNorm operator on the channel statistic vector as AdaShift-**B** (*i.e.*, -**B**asic) and we introduce AdaShift-**MA** that enhances AdaShift-B by exploiting finer-grained tensor-level context cues with a **M**inimalist-style self-**A**ttention operation, which applies LayerNorm operators to calculate Q-K-V attention and removes all the heavy linear projections to preserve the high efficiency of activation. More extensions can be created by varying Δ with finer aggregational operators for tensor-level cues (shown in Section 5.5.3). **Note that** the use of multiple LayerNorm operators on the same input in parallel can be operationally replaced by a learning structure that splices multiple ways of channel-wise scaling operations after a plain LayerNorm operator that removes the parametric (element-wise) affine. *For clearer method description and figure drawing, in this chapter, we use parallel LayerNorm operators to refer to such processing by default.*

From a different perspective, we regard the key of AdaShift as an adaptive fine-grained adjustment of the re-weighting curve ς *w.r.t.* x, hence creating improved ς dynamically, with the incorporated awareness of different ranges of mutual-complementary local and non-local information This avoids the explicit manual modifications to ς, which can be excessively challenging due to the ultra-complexity of underlying mappings.

The contributions of this work are 3-fold:

1. we introduce the novel self-gated neural activation prototype, AdaShift, as an efficiency-boosted solution to the *mismatched feature scoring* problem.
2. based on the AdaShift prototype, we present the efficient practical activation functions, AdaShifts, which improve the prevalent self-gated activation functions significantly and also match/outperform the current SOTA, IIEU(s), with higher efficiency;
3. we extensively validate (a) the effectiveness and versatility of our practical AdaShifts with various vision benchmark datasets; (b) the extensibility and generalizability of our AdaShift prototype with targeted ablation studies and quantitative analysis.

5.2 Related Work

As a max-out approximation to Softplus, ReLU rectified positive and negative inputs by binary masking of 0 and 1, respectively. This paradigm encouraged various derivatives. LeakyReLU [129] suggested a slight leakage factor to the negative interval to make use of negative inputs. PReLU [130] involved negative inputs in parameter updating by an updatable slope. ELU [131] imposes exponential rectifications on negative features. Recent efforts have been taken to develop self-gated-style functions by varying the re-weighting curves ς. As representative methods, SiLU [13] re-weighted features by a Sigmoid function and GELU [12] instead leveraged a Gauss-Error-Function-based (ERF) function to realize finer feature rectifications. Inspired by SiLU, Mish [14] suggested a composite function of Tanh and Softplus. Although demonstrating clear accuracy gains to basic activation functions, typical self-gated functions still found limitations in adaptability.

To compensate for fitting flexibility, SOTA methods introduced auxiliary trainable scaling/bias terms and embedded contextual cues to self-gated activation. Swish [6] extended SiLU by assigning a learnable scaling factor to the input, *i.e.*, $\phi(x) = \varsigma(\kappa x)\, x$, where $\kappa \in \mathbb{R}$. ACON-C [7] further extended Swish by introducing a learnable bound. Meta-ACON [7] enhanced ACON-C by generalizing SE-Net-based [18] channel attention to predict a content-aware input scaling factor. Several SOTA works also investigated new approaches to ERF-based activation. Biswas *et al*. [15] proposed two trainable derivatives of GELU, namely, ErfAct and Pserf, where the former and the later employed exponential and Softplus functions with updatable coefficients to scale the activation inputs, respectively. Encouraged by ACONs, Smooth Maximum Units (*i.e*., SMU-1 and SMU) [8] suggested an ERF-based activation with flexible upper and lower bounds. These new ideas significantly extended the design space of self-gated activation while still leaving the norm-induced *mismatched feature scoring* (Intuition 4.3) problem unsettled, which put a critical constraint on further discriminativeness.

In [2], *i.e*. Chapter 4, we clarified the *mismatched feature scoring* problem and presented IIEU as the initial solution. IIEU was learned with a tailored paradigm to eliminate the norm-induced feature-filter similarity biases by explicit norm-decoupling. This

idea demonstrated SOTA improvements on different networks. However, the direct norm-decoupling on activation inputs likely neutralizes the discriminativeness. This lies in the new observation that feature/filter norms contain informative local details and dataset-level non-local cues for optimizing network parameters. In contrast, our new activation prototype, *AdaShift*, learns discriminative feature activation by comprehensively exploiting local and contextual cues of three different ranges in a particularly simple but effective manner. As a core spirit, unlike IIEU, AdaShift addresses norm-induced *mismatched feature scoring* by temperate dynamic adjustments that evolve a vanilla self-gated ς by adapting to the current learning states. This saves the meaningful norm-related cues and enables AdaShift to improve popular/SOTA activation functions.

5.3 Preliminaries

Our discussion mainly adopts the preliminary settings introduced in Section 4.2, *i.e.*, a set of simple settings with image inputs:

(1) A network includes T sequential learning layers indexed by $\tau = 1, 2, \ldots, T$.

(2) Let $\mathbf{X}^\tau \in \mathbb{R}^{C^\tau \times H^\tau \times L^\tau}$ denotes the input feature map of the layer-τ, where C^τ and $H^\tau \times L^\tau$ show the number of channels and the spatial resolution, respectively.

(3) The learning of the layer-τ at a location $(h, l) \in \Omega_{H^\tau \times L^\tau}$ is denoted by $x_c^{\tau+1}(h, l) \coloneqq \phi\left(\tilde{x}_c^\tau(h, l)\right)$, where $\boldsymbol{w}^\tau(c) \in \mathbb{R}^{C^\tau}$ and $\boldsymbol{x}^\tau(h, l) \in \mathbb{R}^{C^\tau}$ denote the vectorial filter-c and feature $\boldsymbol{x}^\tau(h, l) \in \mathbb{R}^{C^\tau}$, respectively; $\Omega_{H^\tau \times L^\tau}$ represents the spatial lattice of $\mathbf{X}^\tau$ and $\tilde{x}_c^\tau(h, l) = \langle \boldsymbol{w}^\tau(c), \boldsymbol{x}^\tau(h, l) \rangle$ denotes the feature-filter inner product. Note that (a) the layer-τ includes $C_{\tau+1}$ filters; (b) ϕ denotes a given activation function and we rewrite form 5.2 as $\phi\left(\tilde{x}_c^\tau(h, l)\right) = \varsigma\left(\tilde{x}_c^\tau(h, l)\right) \tilde{x}_c^\tau(h, l)$ for clarity (also applicable to prototype 5.2), where ς is the re-weighting function for feature re-calibration.

Note that (1) in discussions of intuitions, we temporarily omit normalization layers (*e.g.*, BatchNorm [127] and LayerNorm [128]) and biases for simplicity (if not specified) and consider them in the formulations of practical methods; (2) for a convolution operation with $K \times K$ field, the supposed settings can be simply met by vectorizing the neighborhood of features/filters to the shape $C^\tau \cdot K^2$ from $C^\tau \times K \times K$. (3) We omit the layer index (*i.e.*, τ) and pixel coordinate (*i.e.*, (h, l)) in the subsequent text for simplified notations. For example, $\boldsymbol{w}^\tau(c)$, $\boldsymbol{x}^\tau(h, l)$, and $\tilde{x}_c^\tau(h, l)$ are denoted by $\boldsymbol{w}$, $\boldsymbol{x}$, and $\tilde{x}$, respectively.

By following the proposed MCDM hypothesis (Chapter 4), we regard (1) a filter $\boldsymbol{w}$ as a learnable ideal candidate which in MCDM [123, 124, 122, 67] denotes the acquirable or virtual optimal decision/choice applied to measure the performance of an alternative candidate by the similarity; (2) a feature vector $\boldsymbol{x}$ as an alternative candidate and its importance score about the corresponding criteria is measured by its similarity to the filter $\boldsymbol{w}$.

5.4 Intuitions and Method

In this section, we first discuss our new intuitions that inspire **AdaShift prototype** and then present two novel **practical AdaShift derivatives** that achieve SOTA improvements over neural activation models with low computational cost, which we refer to as **AdaShift-B** and **AdaShift-MA**, respectively.

5.4.1 AdaShift: Intuitions and Prototype

We begin by clarifying our new intuitions that inspire AdaShift. First, based on the above understanding with the preliminary settings,

(1) we identify that typical self-gated activation functions (*e.g.*, [13, 12, 14]) based on the prototype 5.1 with properties 1 and 2 imply a critical condition that the importance score of a feature vector $\boldsymbol{x}$ about the criteria of a filter $\boldsymbol{w}$ is (strictly) positively correlated to the intensity of the input of activation, *i.e.*, $\tilde{x}$ the feature-filter inner-product. This lies in the fact that their re-weighting functions, *i.e.* ς, are assumed monotonically non-decreasing about $\tilde{x}$.

(2) However, as feature/filter norms can bias the intensity of an inner product as a similarity measure, the implied condition in (1) is likely violated. Therefore, we suppose that the unbiased (*i.e.*, ideal) similarity measure $\varrho(\tilde{x})$ of $\boldsymbol{x}$ to $\boldsymbol{w}$ is not strictly consistent (*i.e.*, increasing) with $\tilde{x}$ over the whole domain.

(3) The analysis in (2) indicates that a basic solution to address *mismatched feature scoring* for self-gated activation is to introduce appropriate ς fully in line with the unbiased similarity measure. However, due to the extreme complexity of underlying mappings of neural learning, the accurate definition of $\varrho(\tilde{x})$ can be excessively difficult.

In Chapter 4, we propose learning to approximate $\varrho(\tilde{x})$ by a tailored learnable prototype, IIEU, leveraging explicit norm-decoupling, *i.e.*, Equation (4.3).Whereas, this paradigm inevitably brings relatively complex gradients, as the (partial) derivative of $s(\boldsymbol{w}) = \frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|}$ *w.r.t.* $\boldsymbol{w}$ is computed by Equation (4.7).

Further, we argue that explicitly decoupling feature and filter norms (*i.e.*, $\|\boldsymbol{x}\|$ and $\|w\|$) from $\tilde{x}$ likely neutralize the discriminativeness of activated features, as we identify

Intuition 5.1. Feature and filter norms present local and dataset-level non-local cues, respectively.

We obtain this intuition by rethinking a common classification process with a Softmax-based classifier Below we introduce our discussion of Intuition 5.1.

Discussion 5.1. We consider a common Softmax-based classification process that takes the vectorial outputs of the classification head (*i.e.*, the last linear layer) as inputs. Let

(1) $\boldsymbol{w}(i) \in \mathbb{R}^C$ denotes a learned filter from the classification head which includes N filters in total, *i.e.* N is the number of classes to categorize and $\boldsymbol{w}(i)$ is learned to represent the class-i;

(2) $\boldsymbol{x} \in \mathbb{R}^C$ denotes a vectorized (*i.e.*, average-pooled) feature inputted to the classification head, served as the learned representation of a raw exemplar (*e.g.*, image);

(3) $\tilde{x}_i = \langle \boldsymbol{w}(i), \boldsymbol{x} \rangle \in \mathbb{R}$ is the corresponding feature-filter inner-products of $\boldsymbol{x}$ and $\boldsymbol{w}(i)$;

(4) $b_i \in \mathbb{R}$ denotes the learned bias term added to the linear projections induced by the filter $\boldsymbol{w}(i)$.

Note that we consider $\forall \boldsymbol{w}(i), \boldsymbol{x}, \boldsymbol{w}(i) \neq \mathbf{0}$ and $\boldsymbol{x} \neq \mathbf{0}$ (*i.e.*, $\|w(i)\| \neq 0$ and $\|x\| \neq 0$) to ensure a meaningful classification. Without loss of generality, let us discuss an assumed case that $\boldsymbol{x}$ is categorized as the class-i. That is, for an arbitrarily given filter $\boldsymbol{w}(j)$ different from $\boldsymbol{w}(i)$, we have the following inequality holds for any $i \neq j$:

$$\begin{aligned} &\frac{e^{\tilde{x}_i + b_i}}{\sum_{c=1}^{C} e^{\tilde{x}_c + b_c}} > \frac{e^{\tilde{x}_j + b_j}}{\sum_{c=1}^{C} e^{\tilde{x}_c + b_c}} \iff e^{\tilde{x}_i + b_i} > e^{\tilde{x}_j + b_j} \\ &\iff e^{\langle \boldsymbol{w}(i), \boldsymbol{x} \rangle + b_i} > e^{\langle \boldsymbol{w}(j), \boldsymbol{x} \rangle + b_j} \\ &\iff e^{\|\boldsymbol{w}(i)\| \|\boldsymbol{x}\| \cos \theta_{\boldsymbol{w}(i), \boldsymbol{x}} + b_i} > e^{\|\boldsymbol{w}(j)\| \|\boldsymbol{x}\| \cos \theta_{\boldsymbol{w}(j), \boldsymbol{x}} + b_j} . \end{aligned} \tag{5.3}$$

Then, as exponential function is monotonically increasing on $\mathbb{R}$, we have inequality 5.3 equivalent to

$$\|\boldsymbol{w}(i)\| \|\boldsymbol{x}\| \cos \theta_{\boldsymbol{w}(i), \boldsymbol{x}} + b_i > \|\boldsymbol{w}(j)\| \|\boldsymbol{x}\| \cos \theta_{\boldsymbol{w}(j), \boldsymbol{x}} + b_j . \tag{5.4}$$

As biases are fixed after learning, let $\alpha = b_j - b_i$, we can rewrite the inequality 5.4 as

$$\|\boldsymbol{w}(i)\| \cos \theta_{\boldsymbol{w}(i), \boldsymbol{x}} - \|\boldsymbol{w}(j)\| \cos \theta_{\boldsymbol{w}(j), \boldsymbol{x}} > \frac{\alpha}{\|\boldsymbol{x}\|} . \tag{5.5}$$

In particular, our major observations from inequality 5.5 are:

1 For the cases where $\|\boldsymbol{x}\| \gg |\alpha|$, *i.e.* $\frac{\alpha}{\|\boldsymbol{x}\|}$ close to 0, the classification of $\boldsymbol{x}$ is (almost) determined by *the filter norms* (e.g., $\|\boldsymbol{w}(i)\|, \forall i \in \{1, 2, \ldots, C\}$) *and norm-decoupled feature-filter similarities,* i.e. *cosine similarities in the discussed case* (e.g., $\cos \theta_{\boldsymbol{w}(i), \boldsymbol{x}}$).

2 For where the feature norms $\|\boldsymbol{x}\|$ and the (absolute intensities of) learned biases $|\alpha|$ are comparable, or $\|\boldsymbol{x}\| \ll |\alpha|$ (hardly exist, as biases are typically small values to avoid neutralizing feature details and over-fittings), the norm-decoupled feature-filter similarities, filter norms, and feature norms are all non-trivial.

3 The norm-decoupled feature-filter similarities and the filter norms are decisive factors to classify $\boldsymbol{x}$, ***regardless of the relative relationship of*** $\|\boldsymbol{x}\|$ and α.

In general, these findings indicate that

1 Filter norms prevalently possess dataset-level non-local cues, and filters leverage these non-trivial context cues to cast significant influences on feature recognitions.

2 The feature norm cues are particularly meaningful when the feature norms are relatively small or close to the learned biases. This attribute induces conditional influences on feature recognition. Intuitively, features with small norms reflect relatively lower confidences/higher uncertainties of identification, therefore feature norms become informative to present private details.

We generalize Intuition 5.1 to common learning layers where the filters are employed to select feature tokens by the feature-filter inner products and identify a promising solution to alleviate norm-induced biases is to cast gentle adaptive adjustments on feature/filter norms, or $\tilde{x}$ them-self since norms are components of $\tilde{x}$. We suppose a key to realizing effective adaptive adjustments is to incorporate ***complementary learning cues*** to compensate for self-gated re-calibration and propose Adashift prototype, *i.e.*,

$$\phi\left(\tilde{x}\right) = \varsigma\left(\tilde{x} + \Delta\right)\tilde{x}\,, \tag{5.6}$$

where Δ defines a learnable shift factor to perform an efficient fine-grained translation on $\tilde{x}$ by exploiting tensor-level context cues; ς denotes a typical self-gated re-weighting function where ***we apply a Sigmoid function by default*** (*i.e.*, the same as SiLU's [13] ς), yet demonstrate wild applicability to various options of ς of different self-gated activation functions (shown in Section 5.5.3). Ensured by the simple prototype, AdaShift is efficient in both inference and gradient calculation, where the (partial) derivative of $\boldsymbol{w}$ is

$$\begin{aligned}\nabla_{\boldsymbol{w}}\phi\left(\boldsymbol{w}\right) &= \frac{\partial\left(\varsigma\left(\tilde{x}+\Delta\right)\tilde{x}\right)}{\partial\boldsymbol{w}} \\ &= \frac{\partial\varsigma\left(\tilde{x}+\Delta\right)}{\partial\left(\tilde{x}+\Delta\right)}\frac{\partial\left(\tilde{x}+\Delta\right)}{\partial\boldsymbol{w}}\tilde{x} + \frac{\partial\tilde{x}}{\partial\boldsymbol{w}}\varsigma\left(\tilde{x}+\Delta\right) \\ &= \varsigma'\left(\tilde{x}+\Delta\right)\tilde{x}\left(\boldsymbol{x} + \frac{\partial\Delta}{\partial\boldsymbol{w}}\right) + \varsigma\left(\tilde{x}+\Delta\right)\boldsymbol{x}\,,\end{aligned} \tag{5.7}$$

where the shift factor Δ is assumed as a function of $\boldsymbol{w}$. Equation (5.7) indicates that AdaShift can work at a low training cost by employing a relatively simple Δ.

We further clarify the intuitions of AdaShift with targeted experiments in Section 5.5.3, where we compare our AdaShift to other relevant prototypes (Section 5.5.3) and SOTA self-gated activation functions built on the modified prototypes of 5.1 (*e.g.*, [7]) with detailed discussions.

5.4.2 Practical Method

We present **AdaShift-B** (-**B**asic) and **AdaShift-MA** (-**M**inimalist **A**ttention) (Figure 5.1) as two examples of practical AdaShifts by embodying the adaptive shift factor Δ with two different efficient designs. AdaShift-B adaptively translates inputs only by leveraging a

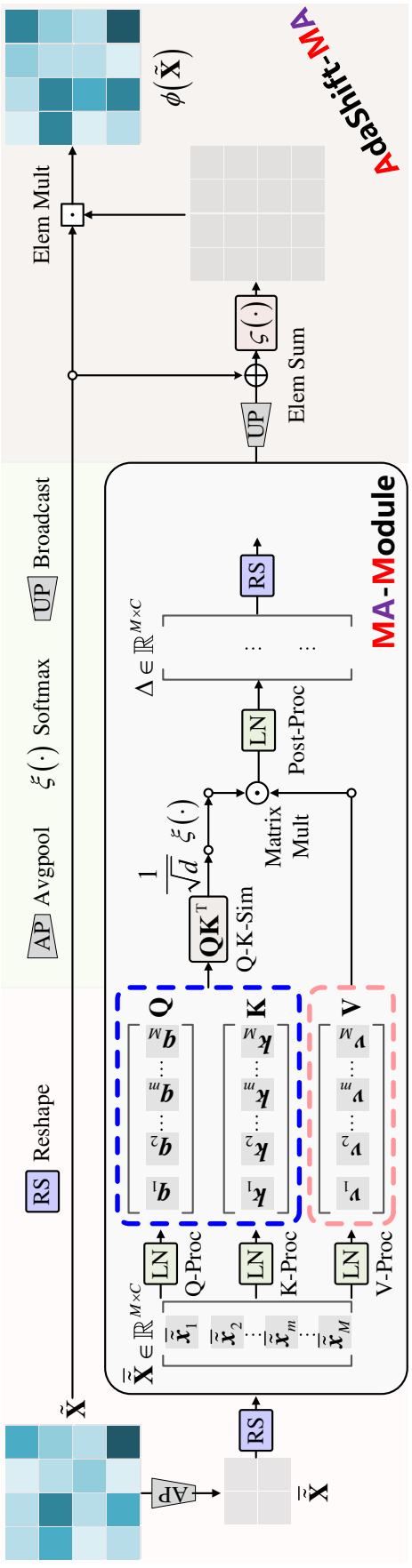


Figure 5.1: Illustration of AdaShift-MA. $M = \lceil H/K_H \rceil \cdot \lceil L/K_L \rceil$. "Elem" denotes "Element-wise" and "Mult" denotes "Multiplication." Note that the use of multiple LayerNorm operators on the same input in parallel can be operationally replaced by a learning structure that splices multiple ways of channel-wise scaling operations after a plain LayerNorm operator that removes the parametric (element-wise) affine.

LayerNorm (LN) to learn tensor-level non-local cues on a global vector of channel statistics. AdaShift-MA further improves AdaShift-B by incorporating finer-grained tensor-level non-local cues, dynamically, with a minimalist self-attention-based module embedded in the re-weighting function ς. Our practical AdaShifts demonstrate SOTA improvements to activation functions with negligible parameters and computational cost.

AdaShift-B. For AdaShift-B, we let Δ be

$$\Delta = \left[\mathrm{LN}\left(\mathrm{avgpool}_{H\times L}\left(\tilde{\mathbf{X}}\right)\right)\right]_c , \tag{5.8}$$

where $\tilde{\mathbf{X}} \in \mathbb{R}^{C\times H\times L}$ denotes the input tensor and c denotes the channel index of $\tilde{x}$ (for alignment); $\mathrm{avgpool}_{H\times L}$ denotes the average-pooling on the global spatial extent $\Omega_{H\times L}$ to generate a vector of channel global statistics $\bar{\tilde{\boldsymbol{x}}} \in \mathbb{R}^C$. LN denotes the LayerNorm to gather tensor-level non-local cues from the channel global statistics.

AdaShift-MA. We propose a minimalist self-attention-based Δ for AdaShift-MA, *i.e.*,

$$\Delta = \left[\mathrm{MA}\left(\mathrm{avgpool}_{K_H\times K_L}\left(\tilde{\mathbf{X}}\right)\right)\right]_c (h_K, l_K) , \tag{5.9}$$

where $\mathrm{avgpool}_{K_H\times K_L}$ denotes a non-overlapped local average-pooling with a kernel-size of $K_H \times K_L$ ($K_H, K_L \in \mathbb{Z}^+$), which produces a patch of channel local statistics $\bar{\tilde{\mathbf{X}}} \in \mathbb{R}^{C\times\left\lceil\frac{H}{K_H}\right\rceil\times\left\lceil\frac{L}{K_L}\right\rceil}$; $(h_K, l_K) = \left(\left\lceil\frac{h}{K_H}\right\rceil, \left\lceil\frac{l}{K_L}\right\rceil\right)$ means the spatial index corresponding to $\tilde{x}$. In particular, MA avoids heavy FLOPs and parameters by replacing all the linear projections with vanilla LayerNorm operations. We suppose this change is feasible for self-gated neural activation, as the core is to mine effective non-local cues to induce dynamic yet gentle adjustments on inputs within the ς.

Enhanced AdaShift-MA Derivatives. We introduce AdaShift as a highly extensible prototype that can flexibly and efficiently leverage various non-local learning modelings of Δ. Based on AdaShift-MA, we further propose three enhanced practical AdaShifts which we refer to as AdaShift-MA-N1, AdaShift-MA-N2, and AdaShift-MA-N3, respectively.

Compared to AdaShift-MA, AdaShift-MA-N1 (operational details are shown in Figure 5.2) jointly attends to the main and the residual features through a united attention process. That is, for a layer that converges the main and the residual features, AdaShift-MA-N1 produces two patches of local channel statistics of the main and the residual features, respectively, and concatenates these two patches along the spatial axis to generate the extended keys and values. The queries are the simple aggregation of the two patches to constrain the complexity. This modification adds zero parameters to AdaShift-MA.

AdaShift-MA-N2 (Figure 5.3) uses a pre-linear-projection before the post-LN to incorporate further fitting flexibility. To avoid excessive parameters, this change is only applied to where the inputs are un-expanded features.

AdaShift-MA-N3 jointly leverages the enhancing strategies of AdaShift-MA-N1 and -N2. It employs AdaShift-MA-N1 at the nodes that converge both the expanded main and

residual features and applies AdaShift-MA-N2 to process unexpanded features only.

Particularly, in Appendix .4, we discuss the attributes of AdaShift-B (as the representative of the practical AdaShift family) in light of the basic intuitions and assumed properties of neural activation inspired by our MCDM hypothesis (introduced in Section 4.3) and show that AdaShift-B is consistent with the basic MCDM-inspired intuitions and holds the corresponding properties.

Note that (1) the above formulations of Δ are tailored to normalized inputs $\tilde{x}$ (*e.g.*, feature-filter inner-products processed by BN or LN), otherwise triggering an imbalanced summation of $\tilde{\mathbf{X}}$ and Δ , since Z-Scoring in a normalization layer (*i.e.*, $\frac{\tilde{\mathbf{X}}-\mu_x}{\sigma_x}$, where μ_x and σ_x are the concerned mean and standard deviation, respectively) actually casts pre-scalings on inputs. We suppose this will impede the effective parameter updating hence resulting in drops in accuracy (discussed in Section 5.5.3). (2) The version of practical AdaShift with tailored modifications for MetaFormer derivatives (*e.g.*, Vision Transformer [19] and ConvNeXt [22]) whose activation inputs are mainly un-normalized (in FFNs) is introduced in Section 5.5.1.

5.5 Experiment

We evaluate the effectiveness and versatility of our practical AdaShifts on various vision benchmark datasets, *i.e.*, ImageNet [86] and CIFAR-100 [28] (image classification); COCO [23] (object detection); KITTI-Materials [1] (RGB road scene material segmentation). Our AdaShift-B and AdaShift-MA are validated by comprehensive experimental comparisons with popular/SOTA activation functions, *i.e.*, (1) Softplus [38], ReLU [5], and ReLU derivatives [131, 130, 129]; (2) popular static self-gated families including [14, 13, 12]; (3) SOTA dynamic self-gated families including [15, 8, 7, 6]; (4) others: [136, 45, 41, 42, 2]. We further validate our AdaShift prototype through extensive ablation studies and analysis of the key observations corresponding to our intuitions and methodological clarifications in Section 5.4.

5.5.1 ImageNet Classification

Implementation details. We evaluate our practical AdaShifts on the popular backbone ResNet [11] of various model sizes, where the baseline networks adopt ReLU as the activation function. For fair comparisons, we adopt the basic CNN training-evaluation protocols [17] (*i.e.*, **cfg-3**, described in 3 of Section 4.5.1) to train each implemented ResNet from scratch.

Main results. We report the comparative results of our AdaShift-B/AdaShift-MA and popular/SOTA activation functions with various networks on ImageNet in Tables 5.1 to 5.4, where our major observations are 3-fold: (1) AdaShift-B enjoys significant improvements over different popular/SOTA activation functions on ResNet backbones of various sizes (except only for its relative MCDM-hypothesis-induced model (on small-size backbones)

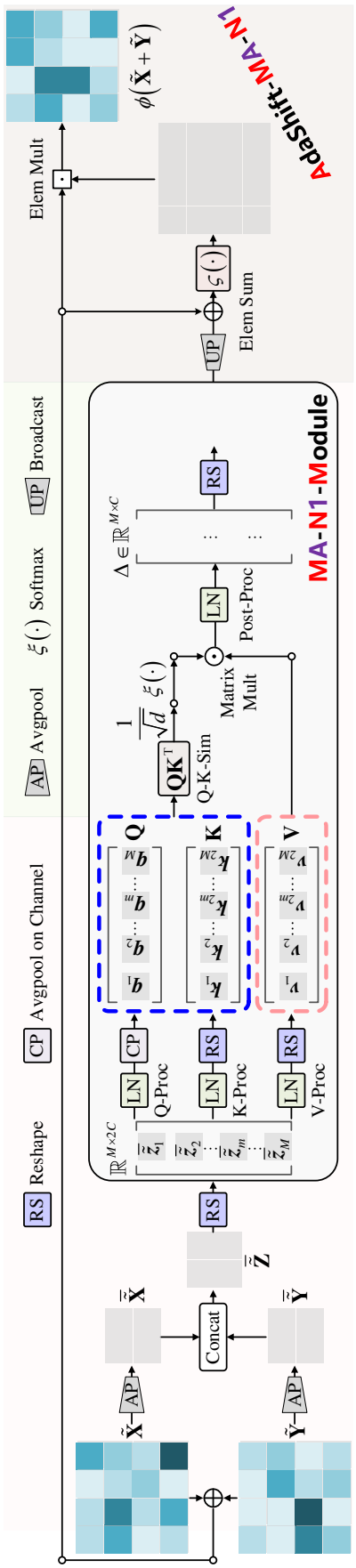


Figure 5.2: Illustration of AdaShift-MA-N1. $M = \lceil H/\kappa_H \rceil \cdot \lceil L/\kappa_L \rceil$. $\bar{\tilde{\mathbf{Y}}} \in \mathbb{R}^{C \times \lceil H/\kappa_H \rceil \times \lceil L/\kappa_L \rceil}$ denotes the residual feature map; $\bar{\tilde{\mathbf{Z}}} = \left[\bar{\tilde{\mathbf{X}}}; \bar{\tilde{\mathbf{Y}}}\right] \in \mathbb{R}^{2C \times \lceil H/\kappa_H \rceil \times \lceil L/\kappa_L \rceil}$ is produced by concatenating the channels of $\bar{\tilde{\mathbf{X}}}$ and $\bar{\tilde{\mathbf{Y}}}$. "Elem" denotes "Element-wise" and "Mult" denotes "Multiplication." Note that AdaShift-MA-N1 is only applicable to the nodes that converge both the main and residual features, otherwise regresses to AdaShift-MA.

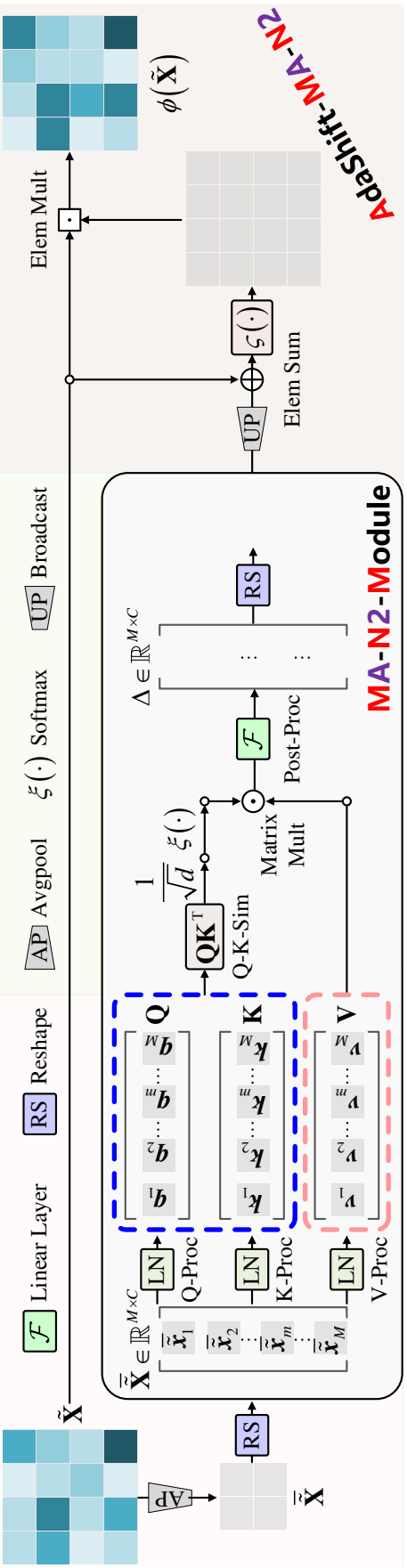


Figure 5.3: Illustration of AdaShift-MA-N2. $M = \lceil H/\kappa_H \rceil \cdot \lceil L/\kappa_L \rceil$. "Elem" denotes "Element-wise" and "Mult" denotes "Multiplication." Note that AdaShift-MA-N2 is only applied to the layers that process unexpanded features (*e.g.*, the second layer of a bottleneck residual block [11]) to avoid bringing excessive parameters.

Table 5.1: Comparison of different activation functions with ResNet-14 [11] backbone on ImageNet. We train each network from scratch with the same training recipes, where "$(+\cdot)$" presents the improvements in Top-1 accuracy of our AdaShift-B and -MA over the ReLU baselines. "NaN" means failed training.

Activation	Backbone	#Params	FLOPs	Top-1 (%)↑
ReLU [5]	ResNet-14 [11]	10.1M	1.5G	68.7
LeakyReLU [129]		10.1M	1.5G	68.8
Softplus [38]		10.1M	1.5G	69.5
ELU [131]		10.1M	1.5G	69.1
GELU [12]		10.1M	1.5G	69.6
SiLU [13]		10.1M	1.5G	69.6
Mish [14]		10.1M	1.5G	69.4
Swish [6]		10.1M	1.5G	69.9
ErfAct [15]		10.1M	1.5G	NaN
Pserf [15]		10.1M	1.5G	69.4
SMU [8]		10.1M	1.5G	70.0
SMU-1 [8]		10.1M	1.5G	68.5
ACON-C [7]		10.1M	1.5G	69.0
Meta-ACON [7]		10.1M	1.5G	70.4
IIEU-B (Chap. 4, [2])		10.1M	1.5G	**73.2**
AdaShift-B (Ours)		10.1M	1.5G	**72.2**(+*3.5*)
AdaShift-MA (Ours)		10.1M	1.5G	**73.9**(+*5.2*)

– our IIEU-B ([2], *i.e.*, Section 4.3.2), *i.e.*, IIEU-B enhances small-size backbones more remarkably and AdaShift-B, in contrast, improves relatively deeper backbones further, a discussion where we qualitatively compare these two relative MCDM-hypothesis-based activation models is included in Appendix .5) and AdaShift-MA boosts AdaShift-B further. Our AdaShifts achieve these large accuracy gains with negligible computational costs and additional parameters to the ReLU baselines which represent the lowest computational cost of neural activation in the comparisons (the practical efficiency analysis can be found in the subsequent text). (2) Compared to the SOTA IIEU-B, AdaShift-B achieves superior accuracies on deep ResNets with clearly higher practical efficiency (measured by *throughput*) by simpler computations (shown in the practical efficiency analysis). This validates the significant applicability and practicality of our AdaShift(s). (3) Enhanced by AdaShifts, networks of relatively small sizes and higher efficiencies can outperform/match the counterparts with far larger scales and deeper layers, *e.g.*, ResNet-50s with AdaShift-B and -MA show remarkable improvements to the large-size ResNet-101 with nearly half the model size and computational cost. These validate our AdaShift for discriminative neural feature activation.

Extending practical AdaShifts with new enhancements.

We further validate the extensibility of the proposed AdaShift prototype with three new

Table 5.2: Comparison of different activation functions with ResNet-26 [11] backbone on ImageNet. We train each network from scratch with the same training recipes, where "$(+\cdot)$" presents the improvements in Top-1 accuracy of our AdaShift-B and -MA over the ReLU baselines.

Activation	Backbone	#Params	FLOPs	Top-1 (%)↑
ReLU [5]	ResNet-26 [11]	16.0M	2.4G	74.9
LeakyReLU [129]		16.0M	2.4G	74.9
Softplus [38]		16.0M	2.4G	75.7
ELU [131]		16.0M	2.4G	75.5
GELU [12]		16.0M	2.4G	75.7
SiLU [13]		16.0M	2.4G	75.8
Mish [14]		16.0M	2.4G	75.8
Swish [6]		16.0M	2.4G	76.1
ErfAct [15]		16.0M	2.4G	75.7
Pserf [15]		16.0M	2.4G	75.7
SMU [8]		16.0M	2.4G	76.1
SMU-1 [8]		16.0M	2.4G	75.1
ACON-C [7]		16.0M	2.4G	75.6
Meta-ACON [7]		16.1M	2.4G	76.5
IIEU-B (Chap. 4, [2])		16.0M	2.4G	**77.7**
AdaShift-B (Ours)		16.0M	2.4G	**77.2**(+*2.3*)
AdaShift-MA (Ours)		16.1M	2.4G	**78.1**(+*3.2*)

practical AdaShift derivatives (introduced in Section 5.4.2), *i.e.*, AdaShift-MA-N1 (Figure 5.2), -N2 (Figure 5.3), and -N3, modified from AdaShift-MA with newly proposed targeted enhancements.

Table 5.5 reports the comparative results of different AdaShift derivatives on ImageNet, where AdaShift-MA-N1, -N2, and -N3 all achieve practical improvements on AdaShift-B and -MA. In particular, AdaShift-MA-N3 demonstrates the superior improvements by merging the enhancing strategies of AdaShift-MA-N1 and -N2. These verify the strong extensibility of our AdaShift prototype.

Practical efficiency.

Despite that SOTA activation functions often add slight parameters and theoretical computational overheads on the ReLU [5] baseline, the actual additional burdens on practical throughput can be more obvious.

We conduct an experimental analysis on the practical efficiency (measured by *throughput*) by comparing our AdaShift-B to SOTA activation functions on ResNet-50 [11] backbone, which includes Swish [6], Pserf [15], SMU [8], and IIEU-B [2] (Section 4.3.2), where we use ReLU as the reference with comparatively highest speed (due to the relatively simplest operations).

We report the comparative practical image *throughput* of the backbone with different

Table 5.3: Comparison of different activation functions with ResNet-50 [11] backbones on ImageNet. We report the implemented results for our AdaShift-B/-MA and the official results for all the other compared models. "N/A" denotes non-applicable/unknown.

Activation	Backbone	#Params	FLOPs	Top-1(%)↑
ReLU [5]	ResNet-50 [11]	25.6M	4.1G	77.2
+SE-Net [18]		28.1M	4.1G	77.8
PReLU [130]		25.6M	4.1G	77.1
PWLU [16]		N/A	N/A	77.8
SMU [8]		25.6M	4.1G	77.5
SMU-1 [8]		25.6M	4.1G	76.9
FReLU [41]		25.7M	4.0G	77.6
DY-ReLU [42]		27.6M	N/A	77.2
ACON-C [7]		25.6M	3.9G	76.8
Meta-ACON [7]		25.8M	3.9G	78.0
IIEU-B (Chap. 4, [2])		25.6M	4.2G	**79.7**
AdaShift-B		25.6M	4.1G	**79.9**(+*2.7*)
AdaShift-MA		25.7M	4.2G	**80.3**(+*3.1*)

activation functions in Table 5.6, where our AdaShift-B adds marginal practical efficiency overhead on the ReLU baseline yet demonstrates competitive speed among SOTAs. It is worth noting that AdaShift-B enjoys significant improvements over SOTA activation functions in accuracy. This validates AdaShift for practical application.

Practical AdaShift for MetaFormer-like network.

In Section 5.4.2, we clarified that AdaShift-B and -MA were tailored to normalized activation inputs (otherwise encountering the *imbalanced summation* problem), hence unsuitable for MetaFormer layers where the activation inputs are commonly un-normalized in **F**eed-**F**orward-**N**etworks (**FFN**s). To this end, we introduce a new practical AdaShift, which

Table 5.4: Comparison of different activation functions with ResNet-101 [11] backbones on ImageNet. We report the implemented results for our AdaShift-B/-MA and the official results for all the other compared models.

Activation	Backbone	#Params	FLOPs	Top-1(%)↑
ReLU [42]	ResNet-101 [11]	44.5M	7.8G	78.9
+SE-Net [18]		49.3M	7.9G	79.3
FReLU [41]		45.0M	7.8G	77.9
ACON-C [7]		44.6M	7.6G	77.9
Meta-ACON [7]		44.9M	7.6G	78.9
IIEU-B (Chap. 4, [2])		44.7M	7.9G	**80.3**
AdaShift-B		44.6M	7.8G	**80.6**(+*1.7*)
AdaShift-MA		44.9M	8.1G	**81.2**(+*2.3*)

Table 5.5: Comparison of ReLU and different practical AdaShift derivatives on ImageNet using ResNet-50 [11] backbone. "AdaShift" is abbreviated by "AdaS."

Activation	Backbone	#Params	FLOPs	Top-1(%)↑
ReLU [5]	ResNet-50 [11]	25.6M	4.1G	77.2
AdaS-B		25.6M	4.1G	79.9
AdaS-MA		25.7M	4.2G	80.3
AdaS-MA-N1	ResNet-50 [11]	25.8M	4.3G	**80.4**
AdaS-MA-N2		28.3M	4.4G	**80.5**
AdaS-MA-N3		28.3M	4.4G	**80.6**

Table 5.6: Evaluation on practical efficiency using ResNet-50 backbone. The image *throughput* is measured on a single RTX A6000 GPU with pure FP32 inputs with a batch size of 128 and image resolution of 224×224. "N/A" denotes non-applicable/unknown (*i.e.*, no accessible official results).

Activation	Backbone	#Params	FLOPs	Top-1(%)↑	Throughput (image / sec.)
ReLU [5]	ResNet-50 [11]	25.6M	4.1G	77.2	**1482.5**
Swish [6]		25.6M	4.1G	77.3	**1476.1**
IIEU-B (Chap. 4)		25.6M	4.2G	**79.7**	1242.6
Pserf [15]		25.6M	4.1G	N/A	1045.3
SMU [8]		25.6M	4.1G	77.5	1046.1
AdaShift-B (Ours)		25.6M	4.1G	**79.9**(*+2.7*)	**1352.8**

we refer to as **AdaShift-MA-X**, modified from AdaShift-MA yet focusing on enhancing MetaFormer-style layers that encode features with FFNs. Specifically, AdaShift-MA-X adds a post-linear-projection at the top of MA-based Δ to avoid imbalanced summation when casting adaptive translations on un-normalized FFN inputs.

By taking into account the computational resource constraints, we choose to evaluate this new AdaShift derivative with ConvNeXt-T [22], a highly efficient advanced ConvNet inspired by MetaFormer architecture [19, 25] with a close size to ResNet-50 [11] but way stronger. In this evaluation, the modified ConvNeXt equipped with our AdaShift-MA-X is compared with the relevant representative MetaFormer counterparts that have close practical efficiency (measured by *throughput*), which includes the original ConvNeXt, **Vi**sion **T**ransformer (**ViT**) [19], PoolFormer [25], and Swin-Transformer [26], where Vision Transformer serves as the baseline. The comparative results are reported in Table 5.7, where the ConvNeXt enhanced by our AdaShift-MA-X enjoys significant improvements in accuracy on the original GELU-based counterpart and other relevant representative MetaFormers activated by GELU function (taking into account the diminishing returns effect on a strong

network backbone).

It is worth noting that **our AdaShift-MA-X, to the best of our knowledge, is the first and only existing activation function that can add practical accuracy gains (*i.e.*, $\geq$ 0.3%) on advanced MetaFormer-like networks** (originally activated by GELU [12] function in common).

This validates the versatility and scalability of our AdaShift prototype.

Table 5.7: ImageNet evaluation of AdaShift-MA-X on ConvNeXt-T [22]. We also introduce three representative vision MetaFormers of close *practical efficiency* as references, *i.e.*, ViT-B/16 [19], PoolFormer-S24 [25], and Swin-Transformer-T [26] (abbreviated by "Swin-Trans-T"), where ViT-B/16 serves as the baseline. The practical image *throughput* is measured on a single RTX A6000 GPU with pure FP32 inputs with a batch size of 128. "$\star$" denotes the improved ViT trained with an extra regularization [27].

Backbone	Activation	#Params	Throughput (image / sec.)	Top-1(%)↑
ViT-B/16$\star$ [19]	GELU [12]	86.6M	775.6	79.7
PoolFormer-S24 [25]	GELU [12]	21.4M	1144.6	80.3
Swin-Trans-T [26]	GELU [12]	28.3M	1052.2	81.3
ConvNeXt-T [22]	GELU [12]	28.6M	1220.1	82.1
	AdaShift-MA-X	32.0M	1075.3	**82.8**

5.5.2 CIFAR-100 Classification

Implementation details. We conduct experimental comparisons of our AdaShift-B and -MA with popular/SOTA activation functions on CIFAR-100 with a public CIFAR version [29] of ResNets which have fewer parameters and computations than the ImageNet network counterparts. For fair comparisons, we train each network from scratch using the standard training recipes [114] (as introduced in Section 4.5.2).

Experimental results. Table 5.8 reports the experimental results, where our **AdaShift-B** and **-MA** improve the popular/SOTA activation functions significantly (except only for its relative model, our IIEU-B, which improves small-size backbones remarkably), which are consistent with the evaluations on ImageNet. These validate the applicability of our AdaShift(s) for datasets of different scales.

5.5.3 Ablation Study

AdaShift prototype.

We explained our main intuitions of AdaShift prototype by rethinking a common Softmax-based classification process in Section 5.4.1. In this ablation study, we further discuss the auxiliary intuitions that help ensure a discriminative feature activation. To this end, we

Table 5.8: Comparison of different activation functions on CIFAR-100. We train each model 8 times and report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1.

Activation	CIFAR-ResNet-29 [11]		CIFAR-ResNet-56 [11]	
	#Params	Top-1(%)↑	#Params	Top-1(%)↑
ReLU [5]	0.3M	70.5±0.3	0.6M	74.4±0.3
ELU [131]	0.3M	72.6±0.2	0.6M	74.7±0.3
PReLU [130]	0.3M	70.1±0.5	0.6M	73.2±0.4
GELU [12]	0.3M	71.4±0.3	0.6M	75.3±0.3
SiLU [13]	0.3M	72.0±0.4	0.6M	75.3±0.4
Swish [6]	0.3M	71.5±0.3	0.6M	74.8±0.2
Mish [14]	0.3M	72.1±0.3	0.6M	75.2±0.3
SMU [8]	0.3M	71.1±0.4	0.6M	74.9±0.3
SMU-1 [8]	0.3M	70.7±0.3	0.6M	74.7±0.2
Pserf [8]	0.3M	71.6±0.2	0.6M	75.3±0.2
ACON-C [7]	0.3M	70.9±0.2	0.6M	74.1±0.3
Meta-ACON [7]	0.3M	72.2±0.3	0.6M	75.7±0.2
IIEU-B (Chap. 4, [2])	0.3M	**74.7**±0.3	0.6M	**77.2**±0.3
AdaShift-B	0.3M	**73.7(+3.2)**±0.4	0.6M	**76.5(+2.1)**±0.3
AdaShift-MA	0.3M	**74.3(+3.8)**±0.3	0.6M	**77.0(+2.6)**±0.4

conduct a tailored experiment and introduce the intuitions from the experimental results and new observations. Specifically, we compare our AdaShift-B with two different sets of targeted control groups:

a A series of **C**ontrol **G**roups (**CG**s) of modified AdaShift-B(s) built on various prospective prototypes of activation functions (*i.e.*, Proto-CG1 to Proto-CG7, as specified in Table 5.9).

b A set of SOTA self-gated activation functions, including ACON-C [7], Meta-ACON [7], SMU-1 [8], and SMU [8], constructed on a specialized self-gated prototype modified from the base self-gated prototype (Equation (5.1)):

$$\phi\left(\tilde{x}\right) = \eta\varsigma\left(\kappa\tilde{x}\right)\tilde{x} + \varepsilon\tilde{x}\,, \tag{5.10}$$

where η, κ, and ε are trainable coefficients (*e.g.*, SMU-1 [8], SMU [8], and ACON-C [7]) or content-aware modules (*e.g.*, Meta-ACON [7]).

In Table 5.9, we report the comparative results of different activation models on CIFAR100 [28] with CIFAR-ResNet-56 [11, 29] backbone, where ReLU serves as the baseline. Note that

(1) all the compared methods in the set-1 use Sigmoid as the ς;

(2) the ς of the SOTA self-gated activation functions in set-2 can be found in Table 5.9;

Table 5.9: Ablation study on different prospective prototypes that apply learnable adjustments and leverage tensor non-local cues, where ReLU is set as the baseline. The experiment is conducted on CIFAR100 [28] with CIFAR-ResNet-56 [11, 29] backbone.

Activation	Prototype	Re-weighting	#Params	Top-1(%)↑
ReLU [5]	—	—	0.6M	74.4±0.3
Proto-CG1	$\phi(\tilde{x}) = \varsigma(\tilde{x})\,\tilde{x}$		0.6M	75.3±0.4
Proto-CG2	$\phi(\tilde{x}) = \varsigma(\kappa\tilde{x})\,\tilde{x}$		0.6M	74.8±0.2
Proto-CG3	$\phi(\tilde{x}) = \varsigma(\Delta\tilde{x})\,\tilde{x}$		0.6M	***73.4±0.3***
Proto-CG4	$\phi(\tilde{x}) = \varsigma(\kappa\tilde{x} + \Delta)\,\tilde{x}$	$\mathrm{sigmoid}(\cdot)$	0.6M	***73.7±0.3***
Proto-CG5	$\phi(\tilde{x}) = \varsigma(\Delta_1\tilde{x} + \Delta_2)\,\tilde{x}$		0.6M	***73.6±0.2***
Proto-CG6	$\phi(\tilde{x}) = \varsigma(\tilde{x} + \Delta)\,(\tilde{x} + \Delta)$		0.6M	75.9±0.3
Proto-CG7	$\phi(\tilde{x}) = \varsigma(\tilde{x} + \Delta_1)\,(\tilde{x} + \Delta_2)$		0.6M	76.2±0.4
ACON-C [7]	$\phi(\tilde{x}) = \eta\varsigma(\kappa\tilde{x})\,\tilde{x} + \varepsilon\tilde{x}$	$\mathrm{sigmoid}(\cdot)$	0.6M	74.1±0.3
Mt-ACON [7]			0.6M	75.7±0.2
SMU-1 [8]		$\mathrm{erf}(\cdot)$	0.6M	74.7±0.2
SMU [8]			0.6M	74.9±0.3
AdaShift-B	$\phi(\tilde{x}) = \varsigma(\tilde{x} + \Delta)\,\tilde{x}$	$\mathrm{sigmoid}(\cdot)$	0.6M	**76.5±0.3**

(3) Δ denotes the proposed shift factor of AdaShift-B;

(4) in particular, Δ_1 and Δ_2 are assigned independently (*i.e.*, employing independent parameters);

(5) κ is specified as channel-wise trainable parameters except for Meta-ACON [7] which learns SE-Net-style [18] channel weights with a lightweight MLP;

(6) CG-1 and CG-2 are equivalent to SiLU [13] and Swish [6], respectively.

Our major observations and the supposed explanations are 4-fold:

a) AdaShift yields the highest accuracy among all the compared prototypes and SOTA self-gated activation models. **This validates the effectiveness of AdaShift prototype.**

b) CG6 which equals to $\phi(\tilde{x}') = \varsigma(\tilde{x}')\,\tilde{x}', \tilde{x}' = \tilde{x} + \Delta$ improves CG1 and CG2 but leads to accuracy drops to AdaShift-B. **This demonstrates that (a) the tensor-level non-local cues are contributing to adaptive feature translations; (b) the *mismatch feature scoring* problem of Act is hard to be eliminated by the direct adjustments on features outside ς and instead, the adaptive adjustments on the re-weighting curve about the input features can be more effective.**

c) CG7 which employs two ways of Δ(s) to shift features from both inside and outside of ς fails to improve AdaShift-B. **This validates that an activation process cannot cumulate the contributions led by the same non-local cues.**

d) CG3, CG4, and CG5 which try to combine channel scalings with the Δ factor fail to achieve practical improvements but demonstrate significant accuracy drops to CG1, CG2, and AdaShift-B. **This indicates that the raw $\tilde{x}$ can serve as an informative**

anchor for Δ to cast tailored adaptive adjustments on the feature re-weighting within the ς. Re-scaling the raw $\tilde{x}$ can be interfering since disrupting the original connections of Δ and $\tilde{x}$.

As auxiliary clarifications of the contributing properties of our AdaShift prototype, we qualitatively compare it with several important control groups that also leverage trainable parameters as follows.

1) **Comparison with Swish [6] (*i.e.*, CG-2).** Swish facilitates a flexible Sigmoid-based re-weighting by adding channel-wise scaling factors on the inputs. However, we identify a critical weakness in the flexibility led by this paradigm. That is, although κ shows no limits on positivity or negativity, it can only be a positive or negative value for an individual channel in an iteration or inference. This attribute constrains Swish-like functions to cast fine-grained adjustments on different feature units within the same channel since we suppose that feature units of different spatial locations can have highly differentiated importance scores as for re-calibrations. Specifically, as for a given channel-c, the re-weighting driven by $\varsigma\left(\tilde{\boldsymbol{X}}_c\right) = \text{sigmoid}\left(\kappa_c \tilde{\boldsymbol{X}}_c\right)$ is still monotonic about the raw input channel slice (*i.e.*, matrix) $\tilde{\boldsymbol{X}}_c \in \mathbb{R}^{H \times L}$, therefore falling short in alleviating the critical *mismatched feature scoring* problem. We suppose this explanation can be generalized to SOTA Swish-like functions that suggest different ς yet demonstrate close results, *e.g.*, ErfAct [15] and Pserf [15] (their results can be found in Table 5.8, which are clearly inferior to our AdaShift-B). In contrast, our AdaShift enables fine-grained flexible adaptive adjustments to different feature units in each individual channel by leveraging an addable translation factor Δ.

2) **Comparison with SOTA self-gated activation functions based on prototype 5.10.** To our understanding, the main change from functions of prototype 5.10 to Swish-like functions is the introducing of a leakage term $\varepsilon\tilde{x}$ which casts feature translations outside the re-weighting curve ς. However, this paradigm remains a critical weakness: The main re-weighting process, *i.e.*, $\eta\varsigma\left(\kappa\tilde{x}\right)$ preserves the (directional) monotonicity on the input $\tilde{x}$ while the leakage term $\varepsilon\tilde{x}$ which translates feature outside ς falls short in making use of the effective nonlinearity, thus resulting in limited feature adjustments.

Based on the above experimental results and qualitative analysis, we summarize our complementary intuitive properties of an effective self-gated neural activation. That is, we expect a self-gated activation function capable of:

(a) casting intense yet flexible changes on the inputs, *i.e.*, capable of giving slight adjustments on the inputs while also capable of drastically varying inputs from positive values to negative depending on the corresponding learning states;

(b) realizing fine-grained adjustments to the inputs, *i.e.*, preserving the diversity of the intensities of feature units while avoiding neutralizing the differences of feature units.

(c) constraining the (main) adjustments within the re-weighting process.

Hypothesis: balanced summation of $\tilde{x}$ and Δ.
We suppose the balanced summation of $\tilde{x}$ and Δ is critical to ensure the effectiveness of AdaShifts (as discussed in Section 5.4.2). To investigate this hypothesis, we compare the original AdaShift-B with three modified AdaShift-B(s) which serve as the targeted control groups: (1) Ada-CG1 which degrades the Δ from $\mathrm{LN}\left(\bar{\tilde{x}}\right)$ to $\gamma\bar{\tilde{x}} + \beta$ by removing the Z-Scoring of LN; (2) Ada-CG2 which replaces the LN in Δ by a linear layer; (3) Ada-CG3, unlike Ada-CG2, which instead applies a linear projection before LN such that the balanced summation is preserved. Table 5.10 reports the comparative results on CIFAR-100 using CIFAR-ResNet-56 backbone, where Ada-CG1 and -CG2 that violate the balanced summation both demonstrate inferior accuracies to the original AdaShift-B. Particularly, although Ada-CG2 leverages a linear layer with considerable extra parameters to impose compensated flexibility to Ada-CG1, it still fails to improve AdaShift-B due to the imbalanced summation. In contrast, Ada-CG3 which saves the balanced summation paradigm achieves meaningful accuracy gains to AdaShift-B. This validates our hypothesis.

Table 5.10: Ablation study on the hypothesis of imbalanced summation of $\tilde{x}$ and Δ, where we report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1.

Activation	Backbone	#Params	FLOPs	Top-1(%)↑
ReLU [5]	CIFAR-ResNet-56 [11]	0.6M	90.7M	74.4±0.3
Ada-CG1		0.6M	91.8M	76.0±0.4
Ada-CG2		1.2M	92.4M	76.3±0.2
Ada-CG3	CIFAR-ResNet-56 [11]	1.2M	92.4M	**77.1±0.3**
AdaShift-B		0.6M	91.8M	**76.5±0.3**

Feature translation w/ or wo/ non-local cues.
We suppose the tensor-level non-local cues incorporated by Δ are the critical complementary information to perform adaptive feature translations. We experimentally investigate this hypothesis by comparing AdaShift-B with two tailored control groups, *i.e.*, modified AdaShift-B(s) (1) removing Δ from the re-weighting process, hence regressing to SiLU [13] (Δ-CG1); (1) leveraging a plain Δ that shifts input features by the trainable channel-wise biases (Δ-CG2). Table 5.11 report the results, where we have two major observations: (1) AdaShift-B enjoys significant improvements to both Δ-CG1 and Δ-CG2; (2) Δ-CG1 and Δ-CG2 demonstrate close accuracies. These validate our hypothesis.

Generalizing AdaShift by varying re-weighting function. We suppose our AdaShift prototype and practical derivatives can be generalized to various options of self-gated re-weighting functions (*i.e.*, ς) different from vanilla Sigmoid function (*e.g.*, ERF-based functions [12]).

For further verification, we hereby investigate the generalizability of our AdaShift prototype about self-gated re-weighting functions with a targeted experiment on CIFAR-100 [28]

Table 5.11: Ablation study on the meaning of non-local cues for Δ. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1 on CIFAR100.

Activation	Backbone	#Params	FLOPs	Top-1(%)↑
ReLU [5]	CIFAR-ResNet-56 [11]	0.6M	90.7M	74.4±0.3
Δ-CG1		0.6M	90.7M	75.3±0.4
Δ-CG2		0.6M	90.7M	75.1±0.4
AdaShift-B	CIFAR-ResNet-56 [11]	0.6M	91.8M	**76.5±0.3**

Table 5.12: Evaluation on the generalizability of AdaShift prototype using different self-gated re-weighting functions $\varsigma(\cdot)$. Activation functions with the suffix "-Ada" denote the modified AdaShift-B(s) that apply the corresponding re-weighting functions.

Activation	Prototype $\phi(\tilde{x})$		Re-weighting	#Params	Top-1(%)↑
	$\varsigma(\tilde{x})\,\tilde{x}$	$\varsigma(\tilde{x}+\Delta)\,\tilde{x}$			
ReLU [5]	—		—	0.6M	74.4±0.3
Tanh	—		—	0.6M	***72.3±0.3***
SiLU [13]	✓		$\mathrm{sigmoid}(\cdot)$	0.6M	75.3±0.4
SiLU-Ada		✓		0.6M	**76.5±0.3**
GELU [12]	✓		$0.5\,(1+\mathrm{erf}\,(\cdot/\sqrt{2}))$	0.6M	75.3±0.3
GELU-Ada		✓		0.6M	**76.3±0.2**
Mish [14]	✓		$\tanh(\mathrm{softplus}(\cdot))$	0.6M	75.2±0.3
Mish-Ada		✓		0.6M	**76.6±0.3**
TanhGate	✓		$0.5\,(\tanh(\cdot)+1)$	0.6M	75.4±0.3
TanhGate-Ada		✓		0.6M	**76.5±0.3**

using CIFAR-ResNet-56 [11, 29], where we compare the modified AdaShift-B(s) employing different ς with the counterparts of original popular/SOTA activation functions that propose/apply the corresponding re-weighting function ς, which include (1) GELU [12] with an ERF-based ς; (2) Mish [14] that suggested $\varsigma(\cdot) = \tanh(\mathrm{softplus}(\cdot))$; (3) a specialized control group, namely, TanhGate which we modified on the vanilla Tanh function, where $\varsigma(\cdot) = 0.5\,(\tanh(\cdot)+1) = {}^{1}/_{1+e^{-2(\cdot)}}$. Note that we use ReLU and Tanh as the baselines and also show the results of the raw AdaShift-B (denoted by SiLU-Ada) with its counterpart model SiLU as a reference group.

Table 5.12 reports the comparative results of different groups of the AdaShift-B derivatives and the counterpart activation functions, where we have two major observations: (1) although differing each other by different ς, the actual performances of the original self-gated activation functions can be indistinguishable; (2) our original and the corresponding modified AdaShift-B(s) improve different self-gated activation function counterparts **significantly and consistently** with only negligible computational cost. These results validate

the generalizability and effectiveness of our AdaShift prototype for discriminative self-gated neural feature activation.

Table 5.13: Comparison of different activation functions on COCO [23] object detection.

Activation	#Params	mAP (%)↑	AP_{50}(%)↑	AP_{75}(%)↑	AP_S(%)↑	AP_M(%)↑	AP_L(%)↑
ReLU [5]	37.7M	36.7	56.0	39.3	21.0	40.2	48.2
IIEU-B (Ch. 4)	37.7M	**38.2**	58.2	40.6	**23.2**	42.1	49.2
SMU [8]	37.7M	37.5	56.6	40.2	21.5	41.5	48.4
Mt-ACON [7]	37.9M	36.5	55.9	38.9	19.9	40.7	**50.6**
Swish [6]	37.7M	37.2	56.3	39.9	21.0	41.1	47.8
AdaShift-B	37.7M	**38.8**	**58.8**	**41.5**	**22.4**	**42.9**	**50.0**

5.5.4 MS COCO Object Detection

Implementation details. We further evaluate the generalizability and versatility of AdaShift (using AdaShift-B) on MS COCO [23] object detection by comparing it with the ReLU [5] baseline and other popular/SOTA activation functions, *i.e.*, Meta-ACON [7], SMU [8], IIEU-B [2] (Section 4.3.2), Swish [6]. For fair comparisons, we apply the default implementation procedure ($1\times$ schedule) in MMDetection toolbox [139] and report the results on the standard evaluation metrics, *i.e.*, mAP as the primary metric of averaged precisions and AP_{50}, AP_{75}, AP_S, AP_M, AP_L as the specific APs at different scales. We use a popular efficient detector RetinaNet that extracts feature maps with ResNet-50 encoders equipped with different activation functions, each of which is applied with their corresponding ImageNet pre-trained weights. By following the common practice, we ensure reproducibility by keeping on the deterministic mode in each implementation. Note that we report the official results for Meta-ACON as our re-implemented results are lower, possibly led by the different implementation environments.

Experimental results. The experimental results are shown in Table 5.13, where our AdaShift-B achieves significant gains in accuracy over different popular/SOTA activation functions. It is worth noting that the highly consistent and significant improvements over the baseline and popular/SOTA activation functions on various vision benchmarks verify the strong versatility and generalizability of AdaShift for effective self-gated neural activation.

5.5.5 KITTI-Materials Road Scene Material Segmentation

Implementation details. We further validate the generalizability of our AdaShift by comparing AdaShift-B with popular/SOTA activation functions on KITTI-Materials [1] RGB road scene material segmentation. For fair comparisons, we adopt the official training and evaluation configures [1] with a common framework composed of ResNet-50 encoder [11] and the multi-level All-MLP segmentation head [4].

Table 5.14: Comparison of popular/SOTA activation functions on KITTI-Materials [1] road scene material segmentation.

Activation Function	Encoder	SegHead	#Params	mIoU(%)↑
ReLU [5]	ResNet-50 [11]	All-MLP [4]	31.7M	40.2
Meta-ACON [7]			31.9M	41.7
IIEU-B (Chap. 4)			31.7M	**42.4**
SMU [8]			31.7M	40.6
Swish [6]			31.7M	41.2
AdaShift-B (Ours)			31.7M	**42.2**

Experimental results. Table 5.14 demonstrates the comparative results of different popular/SOTA activation functions and the ReLU baseline, where AdaShift-B (1) achieves matchable accuracy to its relative activation model IIEU-B (Section 4.3.2) with clearly higher efficiency (as demonstrated in Table 5.6, Section 5.5.1); (2) outperforms other SOTA activation functions by a clear margin.

This further verifies the versatility and generalizability of our AdaShift.

5.6 Summary

In this Chapter, we propose to learn discriminative self-gated neural feature activation with a novel AdaShift prototype inspired by the new intuitions of feature-filter context in neural learning. AdaShift adaptively translates the activation inputs by comprehensively exploiting informative local and non-local cues of different ranges, therefore performing fine-grained adjustments to the feature re-weighting in a particularly simple yet effective manner. Built on the new prototype, our practical AdaShifts significantly improve popular/SOTA activation functions on various vision benchmarks with only negligible computational cost and parameters added to ReLU baseline.

Chapter 6

Conclusion and Future Directions

6.1 Conclusion

In this dissertation, we extensively investigated two significant challenges in deep-learning-based visual recognition, *i.e.*, (1) a special task that is unexplored yet of particular meaning for general scene understanding – RGB road scene material segmentation; (2) a general problem that touches one of the foundations of neural networks and neural representations, *i.e.*, *neural feature activation with image inputs*. We present three novel methods by leveraging self-attention and self-gating mechanisms, lay in the very original intention of neural attention and gating, *i.e.* discriminative yet low-cost.

First, as presented in Chapter 3, we address RGB **R**oad scene **M**aterial **S**egmentation (**RMS**), an emerging avenue of visual recognition problem, based on the new benchmark, KITTI-Materials, by proposing a novel self-attention-based framework that effectively fuses texture and contextual cues. The framework, *i.e.*, RMSNet, achieves this with SAMixer, a novel model that performs effective yet highly efficient multi-level multi-scale feature fusion with the tailored MSA mechanism, built on a newly derived balanced Q-K-Sim measure and BLSED strategy. Extensive experimental evaluations and ablation studies on KITTI-Materials dataset validate the effectiveness and scalability of our model designs.

Through this new research, we have not only gained new knowledge of RGB RMS and scene understanding but also found new intuitions for understanding the mechanism of neural activation.

Despite the significant differences in attributes of the *material* categories and the *object* categories, we aim to explore the important commonality for deep-learning-based visual recognition. Motivated by this, our observation that *the same road scene image can naturally have different annotations of object categories and material categories, simultaneously*, inspires our new intuitions of modeling adaptive information selection on neural features for general visual recognition.

More specifically, for supervised learning, we suppose a neural network encodes the discriminative representations through *learning to select the targeted information corresponding to the designated annotations (ground-truths)*. We believe a nonlinear neural activation process that incorporates inductive bias to help fit the underlying mappings of objectives is a key to the oriented knowledge selection in a neuron. Encouraged by this, as presented in Chapter 4, we propose to interpret neural feature activation from the new perspective of multi-criteria decision-making, where we identify the critical yet unstudied problem, *mismatched feature scoring*, and present our activation model IIEU as the initial solution originated from our new intuitions of effective neural feature activation. We validate our new intuitive hypotheses and the novel practical methods inspired by them, *i.e.*, IIEUs, through comprehensive experimental analysis and extensive comparisons with popular and SOTA activation models on various vision benchmark datasets, where IIEUs achieve the new SOTA improvements to the ReLU baseline and significantly outperform the current prevailing and SOTA activation models.

Then, as presented in Chapter 5, we propose to learn discriminative self-gated neural feature activation with the novel AdaShift prototype inspired by the original MCDM hypothesis with new intuitions of understanding the feature-filter context in neural learning. Our AdaShift prototype adaptively translates the activation inputs by comprehensively exploiting informative local and non-local cues of different ranges, therefore performing fine-grained adjustments to the feature re-weighting in a particularly efficient yet effective manner. Built on the AdaShift prototype, our practical AdaShifts significantly improve popular/SOTA activation functions on various vision benchmarks with high efficiency.

Furthermore, in Appendix .6, we demonstrate the significant potential of our MCDM hypothesis for interpreting the working mechanism of neural activation. To the best of our knowledge, our MCDM hypothesis is the first and only specialized hypothesis for predictive qualitative assessments of neural activation models, which is more concrete and effective than the general constraints that have been adopted in the analysis of activation functions, *e.g.*, *nonlinearity* and *Lipschitz continuity*.

We believe that our three works expand the avenues of deep-learning-based visual recognition research and contribute to the understanding of self-gated and self-attended neural feature selection.

6.2 Future Directions

Finally, we discuss the unexplored problems related to our current works and will leave them for future research.

Unified MCDM-inspired self-gated neural activation model.
Based on the MCDM hypothesis, we propose IIEU and AdaShift as two different solutions to the *mismatched feature scoring* problem (Intuition 4.3) and introduce new improvements to neural activation. However, these two neural activation models present notable differences in attributes of activation, where we find IIEU enhances small-size backbones more remarkably while AdaShift introduces further improvements on deeper backbones with higher practical efficiency. As discussed in Section .5, we suppose such differences primarily stem from their strategies in addressing the *mismatched feature scoring* problem, where IIEU eliminates the possible norm-based biases through a straightforward norm-decoupling scheme for input features and filters while in contrast, inspired by the new intuitions obtained from Softmax-based classification, AdaShift introduces relatively gentle adjustments on the inputs by exploiting different ranges of local and non-local cues jointly and adaptively in the re-weighting process. This motivates us to investigate and develop a possible approach to unify the differentiated strengths of these two activation prototypes. New intuitions and/or hypothetical properties may be required.

Currently, we have started this study and achieved initial progress by deriving a new activation model (unnamed yet) that improves IIEU and AdaShift in preliminary experiments (on CIFAR-100 dataset). In future work, we plan to refine the analysis of the current model and explore further practical improvements based on the new analyses and intuitions (new models different from the current one may also be proposed).

Hypothetical neural activation properties involving filter updating.
The current MCDM-hypothesis-based properties of neural activation (for visual recognition) are still primitive and mainly assess activation models (functions) from the aspect of feature inference which is relatively simpler to analyze than from filter updating. However, there remain many phenomena of activation that cannot be explained or predicted by the current hypothesis. For example, in Table 3, $\left\{\phi_i^{(4)}\left(\tilde{x}\right)\right\} \mid i = 1, 2, 3$ violate the hypothetical property Property 4.2 (CNI) by the same way and are expected to have similar extent of decreases in accuracy. However, experimental evaluation demonstrates that $\phi_2^{(4)}$ and $\phi_3^{(4)}$ both led to failed training while $\phi_1^{(4)}$ did not (although also yielded bad performance as predicted).

To improve the current MCDM hypothesis, I plan to explore new/improved hypothetical properties of neural activation from the aspect of filter updating.

Comprehensive filter-filter relationship modeling in neural networks.
The proposed MCDM hypothesis is essentially an intuitive generalization of MCDM process to feature-filter relationships in neural learning. This implies an unexplored field (to

the best of my knowledge) of extending the design space of neural networks by realizing effective filter-filter relationship modeling.

In future work, I plan to investigate this new avenue based on the existing references of comprehensive feature-feature relationship modeling (*e.g.*, self-/cross-attention mechanisms and global filtering).

Neural activation with non-function mappings.

Our work in Chapter 4 implies the potential of leveraging non-function mappings (about the input $\tilde{x}$) for effective neural activation. This stem from a fundamental intuitive assumption of MCDM hypothesis, *i.e.*, different inputs (*e.g.*, two feature-filter inner products $\tilde{x} = \langle \boldsymbol{w}, \boldsymbol{x} \rangle, \tilde{y} = \langle \boldsymbol{v}, \boldsymbol{y} \rangle \in \mathbb{R}$) with the same value (*i.e.*, $\tilde{x} = \tilde{y}$) possibly have different importance scores measured based on the unbiased feature-filter similarities. More abstractly, based on this assumption, an activation model (a mapping) with an input $\tilde{x}$ (a given pre-image) may output more than one result (images), this spontaneously induces a non-function mapping.

In future work, we plan to investigate this unstudied avenue by proposing targeted hypotheses and developing new methods.

Neural activation model for numerical regression problems.

The current MCDM hypothesis is based on settings of image-based visual recognition and may be non-applicable to numerical regression problems.

In future work, we plan to develop targeted hypotheses and methods of effective neural activation for numerical regression problems.

Appendix .1 Discussions, Deductions, and Proofs for Section 4.3.1

.1.1 Proof of Proposition 4.1

In Section 4.3.1, we introduce Proposition 4.1 based on Intuition 4.2 and Property 4.1.

Retrospect.
For simpler notations, following we denote $\varrho(\tilde{x})$ as ϱ_x (*i.e.*, $\varsigma(\varrho(\tilde{x}))$ is denoted by $\varsigma(\varrho_x)$).

Property 4.1. $|\varsigma(\varrho_x)| \geq |\varsigma(\varrho_y)|$ if $\varrho_x \geq \varrho_y$.

Where $\varsigma(\varrho_x)$ is continuous and differentiable about ϱ_x on the domain (or at most has finite points where the left- and right-hand limits of the function exist but are unequal). Note that Property 4.1 is ensured by ς, as the monotonicity of $|\varrho_x|$ about ϱ_x is uncertain. Moreover, Property 4.1 can be met with the more specific conditions, *i.e.*,

Proposition 4.1. Property 4.1 $\Longleftrightarrow$ (1) $\varsigma(\varrho_x)$ is monotonically increasing (non-decreasing) about $\varrho_x \wedge \varsigma(\varrho_x) \geq 0 \vee$ (2) $\varsigma(\varrho_x)$ is monotonically decreasing (non-increasing) about $\varrho_x \wedge \varsigma(\varrho_x) \leq 0$ ($\wedge$ and $\vee$ denote logical "and" and "or," respectively).

In particular, as for the cases $\varsigma(\varrho_x) \geq 0$ and $\varsigma(\varrho_x) \leq 0$ which are symmetrical about $\varsigma(\varrho_x) = 0$ and mutually exclusive with each other excluding $\varsigma(\varrho_x) = 0$, the former can be easily extended to the latter once proven and vice versa.

Proposition $\Longrightarrow$. Property 4.1 $\Longrightarrow$ (1) $\varsigma(\varrho_x)$ is monotonically increasing (non-decreasing) about $\varrho_x \wedge \varsigma(\varrho_x) \geq 0 \vee$ (2) $\varsigma(\varrho_x)$ is monotonically decreasing (non-increasing) about $\varrho_x \wedge \varsigma(\varrho_x) \leq 0$.

Proof. First, we assume that we can find a $\varsigma(\varrho_x) > 0$ and $\varsigma(\varrho_y) < 0$, simultaneously. As such, our goal is to find a paradox with this assumption.

With the prerequisite condition: $\varsigma(\varrho_x)$ is continuous about ϱ_x, $\forall\varrho_x$ and the assumed condition: $\exists\varrho_x, \varrho_y$ such that $\varsigma(\varrho_x) > 0, \varsigma(\varrho_y) > 0$, suppose $(\varrho_z, \varsigma(\varrho_z)) : \varrho_x > \varrho_z > \varrho_y$ is a moving point between $(\varrho_y, \varsigma(\varrho_y))$ and $(\varrho_x, \varsigma(\varrho_x))$, then, $(\varrho_z, \varsigma(\varrho_z))$ traverses through the point $(\varrho_{z_0}), 0$ and we have:

$$\varsigma(\varrho_x) \geq |\varsigma(\varrho_{z_0})| = 0 \geq |\varsigma(\varrho_y)| = -\varsigma(\varrho_y) \Longrightarrow |\varsigma(\varrho_y)| = 0\,. \tag{1}$$

But this deduced conclusion leads to a paradox to the assumption: $\varsigma(\varrho_y) < 0$, so we cannot find such a ϱ_y and $\varsigma(\varrho_y)$.

Besides, it can be deduced that both the cases $\exists\varsigma(\varrho_x) > 0, \varsigma(\varrho_y) = 0$ and $\exists\varsigma(\varrho_x) = 0, \varsigma(\varrho_y) < 0$ does not lead to paradoxes. That is, with the above deductions, we have $\forall\varrho_x, \varsigma(\varrho_x) \geq 0 \vee \varsigma(\varrho_x) \leq 0$.

Next, we first consider the condition: $\varsigma(\varrho_x) \geq 0$. Then, Property 4.1 can be specified to: $\forall\varrho_x, \varrho_y$ in the domain, $|\varsigma(\varrho_x)| = \varsigma(\varrho_x) \geq \varsigma(\varrho_y) = |\varsigma(\varrho_y)|$ if $\varrho_x \geq \varrho_y$. Therefore, Property 4.1 is monotonically increasing (*i.e.*, non-decreasing) about ϱ_x, $\forall\varrho_x$.

Similarly, with the condition: $\varsigma(\varrho_x) \leq 0$, Property 4.1 can be specified to: $\forall \varrho_x, \varrho_y$ in the domain, $|\varsigma(\varrho_x)| = -\varsigma(\varrho_x) \geq -\varsigma(\varrho_y) = |\varsigma(\varrho_y)|$ if $\varrho_x \geq \varrho_y$, *i.e.*, $\varsigma(\varrho_x) \leq \varsigma(\varrho_y)$. Therefore, Property 4.1 is monotonically decreasing (non-increasing) about ϱ_x, $\forall \varrho_x$.

This completes the proof. ■

Proposition $\Longleftarrow$. Property 4.1 $\Longleftarrow$ (1) $\varsigma(\varrho_x)$ is monotonically increasing (non-decreasing) about $\varrho_x \wedge \varsigma(\varrho_x) \geq 0 \vee$ (2) $\varsigma(\varrho_x)$ is monotonically decreasing (non-increasing) about $\varrho_x \wedge \varsigma(\varrho_x) \leq 0$.

Proof. With the condition (1): $\forall \varrho_x, |\varsigma(\varrho_x)| = \varsigma(\varrho_x)$ and $\varsigma(\varrho_x)$ is monotonically increasing (non-decreasing) about ϱ_x, we have: $\forall \varrho_x, \varrho_y$ in the domain, $|\varsigma(\varrho_x)| = \varsigma(\varrho_x) \geq \varsigma(\varrho_y) = |\varsigma(\varrho_y)|$ if $\varrho_x \geq \varrho_y$. This ensures the Property 4.1.

Similarly, with the condition (2), we have: $|\varsigma(\varrho_x)| = -\varsigma(\varrho_x) \geq -\varsigma(\varrho_y) = |\varsigma(\varrho_y)|$ if $\varrho_x \geq \varrho_y$. This ensures the Property 4.1.

This completes the proof. ■

Summary. We complete the proofs for both the partial propositions (*i.e.*, directions "$\Longrightarrow$" and "$\Longleftarrow$") of Proposition 2, which ensures Proposition 2.

.1.2 Property 4.2 and Property 4.3

Below we introduce the strict cases of Property 4.2 and Property 4.3 based on Intuition 4.4 and Intuition 4.5, respectively, for where $\varrho(\tilde{x})$ is (a) (uniformly) continuous about $\tilde{x}$ on the domain; (b) differentiable about $\tilde{x}$ or at most has a finite number of points where the left- and right-hand limits exist but are unequal.

Property .1. ***(CNI)*** Strict case: $\exists \eta \in \mathbb{R}, \mathcal{M}_{x^-} \geq 0$ such that $\forall \varrho(\tilde{x}) < \eta$, we have $|\rho(\tilde{x})\,\tilde{x}|\,|_{\varrho(\tilde{x})<\eta} \leq \mathcal{M}_{x^-}$.

Property .2. ***(PPI)*** Strict case: $\exists \eta \in \mathbb{R}, \mathcal{M}_{x^+} \geq 0$, such that $\forall \varrho(\tilde{x}) > \eta$ we have $|\nabla_{\tilde{x}}(\rho(\tilde{x})\,\tilde{x})|\,|_{\varrho(\tilde{x})>\eta} \leq \mathcal{M}_{x^+}$ at any $\tilde{x}$ where $\phi(\tilde{x})$ is differentiable.

In particular, we summarize a simple condition to ensure the Property .1, *i.e.*,

Proposition .1. $\lim_{\varrho(\tilde{x})\to-\infty}(\varsigma(\varrho(\tilde{x}))\,\tilde{x}) = 0 \Longrightarrow$ *Property .1 (*i.e.*, strict case of Property 4.2).*

Proof.

a. Simple case. First, we consider the simple case where $\varsigma(\varrho(\tilde{x}))$ is fully continuous and differentiable about $\varrho(\tilde{x})$.

Then, as we have the pre-condition: $\varrho(\tilde{x})$ is continuous and differentiable about $\tilde{x}$ on $\mathbb{R}$ (mentioned in Property 4.1), for $\forall \tilde{x} \in [a, b]$, where $a, b \in \mathbb{R}$ and $[a, b]$ an arbitrary finite interval, then, $\varrho(\tilde{x})$ and $\varsigma(\varrho(\tilde{x}))$ are bounded, simultaneously.

Then, because $\varsigma(\varrho(\tilde{x}))$ and $\tilde{x}$ are both bounded on $\tilde{x} \in [a, b]$, we have $|\varsigma(\varrho(\tilde{x}))\,\tilde{x}|$ bounded. As the upper-bound of $|\varsigma(\varrho(\tilde{x}))\,\tilde{x}|$ exists, then, without loss of generality, let $\mathbb{M}_{x^-}$ denote the set of the upper-bound, such that we have $\mathcal{M}_{x^-} \in \mathbb{M}_{x^-}$. That is, $\exists \mathcal{M}_{x^-}$

such that $|\phi(\tilde{x})| \leq \mathcal{M}_{x^-}$. As such, the conclusion: $|\rho(\tilde{x})\,\tilde{x}|\,|_{\varrho(\tilde{x})<\eta} \leq \mathcal{M}_{x^-}$ holds as long as $|\varsigma(\varrho(\tilde{x}))\,\tilde{x}|$ is upper-bounded when $\varrho(\tilde{x}) < \eta$.

With the above deduction, that $|\varsigma(\varrho(\tilde{x}))\,\tilde{x}|$ is unbounded is only possible when $\varrho(\tilde{x})$ approaches $-\infty$ where $\tilde{x}$ approaches to $-\infty$ or $+\infty$. Note that now the direction is unknown. But, as the given condition $\phi(-\infty) = 0$ constraints that:

$$\lim_{\varrho(\tilde{x})\to-\infty} |\varsigma(\varrho(\tilde{x}))\,\tilde{x}| = 0\,, \tag{2}$$

then, we have $|\varsigma(\varrho(\tilde{x}))\,\tilde{x}|$ bounded, no matter $\tilde{x}$ approaches to $-\infty$ or $+\infty$. That is, we have $|\rho(\tilde{x})\,\tilde{x}|\,|_{\varrho(\tilde{x})<\eta} \leq \mathcal{M}_{x^-}$.

This completes the proof. ■

b. Extended case. Here, we discuss the extended case: $\varsigma(\varrho(\tilde{x}))$ is fully continuous about $\varrho(\tilde{x})$ while has a finite number of non-differentiable points where the corresponding left-hand and right-hand limits exist but are unequal.

As the left-hand and right-hand limits always exist for any point on $\varsigma(\varrho(\tilde{x}))$, the boundedness of $\varsigma(\varrho(\tilde{x}))$ is ensured at any finite interval. That is, like in the fully continuous case, the conclusion is only possible to be violated when $\varrho(\tilde{x})$ approaches $-\infty$.

But, because the number of the non-differentiable points is finite and the continuity always holds, we can still find such a η which is smaller than all the $\varrho(\tilde{x})$ where $\varsigma(\varrho(\tilde{x}))$ are non-differentiable but both the corresponding left-hand and right-hand limits exist. Therefore, the proof of the case **a** can be directly generalized to the case **b**.

This completes the proof. ■

Further discussion. As a corollary related to Proposition .1, we identify a more specific condition to ensure the strict case of Proposition .1. This specific condition is easier to apply to help the design of activation models, which we suggest as: (1) $\varsigma(-\infty) = 0$; (2) $\exists\eta_\varrho \in \mathbb{R}$ and $\forall k \in \mathbb{R}$, if $|\varsigma(\varrho(\tilde{x}))| \leq \left|\frac{k}{\tilde{x}}\right|$ holds for $\forall\varrho(\tilde{x}) < \eta_\varrho$.

That is, in intuition, we suppose as long as the (absolute) reweighting function $|\varsigma(\varrho(\tilde{x}))|$ can be upper-bounded by the simple reference function(s) $\left|\frac{k}{\tilde{x}}\right|$ when the ideal similarity $\varrho(\tilde{x})$ gradually approaches to $-\infty$, the Proposition .1 holds.

Proof. As discussed in the proofs of the cases **a** and **b**, the boundedness of $|\phi(\tilde{x})|$ is only possible to be violated when $\varrho(\tilde{x})$ approaches $-\infty$ where $\tilde{x}$ approaches to $-\infty$ or $+\infty$. Then, as we have the condition: $|\varsigma(\varrho(\tilde{x}))| \leq \left|\frac{k}{\tilde{x}}\right|$ for $\forall\varrho(\tilde{x}) < \eta_\varrho$, we have:

$$\begin{aligned} |\varsigma(\varrho(\tilde{x}))\,\tilde{x}|\,|_{\varrho(\tilde{x})<\eta_\varrho} &= |\varsigma(\varrho(\tilde{x}))|\,|\tilde{x}|\,|_{\varrho(\tilde{x})<\eta_\varrho} \\ &\leq \left|\frac{k}{\tilde{x}}\right| |\tilde{x}|\,|_{\varrho(\tilde{x})<\eta_\varrho} = |k|\,|_{\varrho(\tilde{x})<\eta_\varrho}\,, \end{aligned} \tag{3}$$

where $\lim_{\varrho(\tilde{x})\to-\infty} |k| = |k|$. That is, $|\varsigma(\varrho(\tilde{x}))\,\tilde{x}|$ is bounded.

Therefore, we complete the proof. ■

Summary. We complete the proofs for the cases **a** and **b** of Proposition .1, which ensures Proposition .1. We further the discussion to a more specific condition that we find easier to apply to help the design of neural feature activation models.

.1.3 Discussion on Equation (4.3)

In particular, we treat and analyze IIEU prototype (Equation (4.3)) as a function (instead of a non-function mapping) of $\tilde{x}$ (where $\tilde{x} = \langle \boldsymbol{w}, \boldsymbol{x} \rangle$) when discussing feature inference, despite that the original *cosine similarity* (*i.e.*, $\frac{\tilde{x}}{\|\boldsymbol{x}\|\|\boldsymbol{w}\|}$) defines a non-function mapping of $\tilde{x}$, where the (two) reasons are clarified below Equation (4.3). Following is a formalized discussion with the simple yet reasonable assumed settings.

Preliminaries.
Assume $\mathbf{w} \in \mathbb{R}^C$ and $\mathbf{x} \in \mathbb{R}^C$ ($C \geq 2$) are two **independently and identically distributed** vector-valued random variables, they share the same multivariate Gaussian distribution, *i.e.*, $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \boldsymbol{\Sigma} \neq \mathbf{0}$). Let $\left\{ \boldsymbol{w}(m) \mid \boldsymbol{w} \in \mathbb{R}^C; m = 1, \ldots, M, M \in \mathbb{Z}^+, M \geq 2 \right\}$ and $\left\{ \boldsymbol{x}(n) \mid \boldsymbol{x} \in \mathbb{R}^C; n = 1, \ldots, N, N \in \mathbb{Z}^+, N \geq 2 \right\}$ denote the sets of M and N times of random sampling (*i.e.*, finite times of observation) of $\mathbf{w}$ and $\mathbf{x}$, respectively, which also represent filters and features. Let $\tilde{x}_{m,n} = \langle \boldsymbol{w}(m), \boldsymbol{x}(n) \rangle$ and $(a_{m,n})_{M \times N}$ denotes the matrix of feature-filter inner products (real values), *i.e.*, $\tilde{x}_{m,n} = a_{m,n}$ (*i.e.*, a known real value).

Discussion.
With the assumed settings above, the expectation and variance of the distribution of $\tilde{\mathrm{x}} = \langle \mathbf{w}, \mathbf{x} \rangle$ can be calculated as:

$$\mathbb{E}\left[\langle \mathbf{w}, \mathbf{x} \rangle\right] = \sum_{c=1}^{C} \mathbb{E}\left[\mathrm{w}_c \mathrm{x}_c\right] = \sum_{c=1}^{C} \mathbb{E}\left[\mathrm{w}_c\right] \mathbb{E}\left[\mathrm{x}_c\right] = \sum_{c=1}^{C} \mu_c^2 = \langle \mu, \mu \rangle = \|\mu\|^2 , \tag{4}$$

and,

$$\begin{aligned}
\mathrm{Var}\left(\langle \mathbf{w}, \mathbf{x} \rangle\right) &= \sum_{c=1}^{C} \mathrm{Var}\left(\mathrm{w}_c \mathrm{x}_c\right) \\
&= \sum_{c=1}^{C} \Big(\mathrm{Var}\left(\mathrm{w}_c\right) \mathbb{E}\left[\mathrm{x}_c\right]^2 + \mathbb{E}\left[\mathrm{w}_c\right]^2 \mathrm{Var}\left(\mathrm{x}_c\right) + \mathrm{Var}\left(\mathrm{w}_c\right) \mathrm{Var}\left(\mathrm{x}_c\right) \Big) \\
&= \sum_{c=1}^{C} \left(\Sigma_{c,c} \mu_c^2 + \mu_c^2 \Sigma_{c,c} + \Sigma_{c,c} \Sigma_{c,c} \right) \\
&= \sum_{c=1}^{C} \left(2 \Sigma_{c,c} \mu_c^2 + \Sigma_{c,c}^2 \right) \\
&= 2 \sum_{c=1}^{C} \Sigma_{c,c} \mu_c^2 + \sum_{c=1}^{C} \Sigma_{c,c}^2 ,
\end{aligned} \tag{5}$$

respectively, *i.e.*, $\tilde{\mathrm{x}} \sim \mathcal{N}\left(\|\mu\|^2, 2\sum_{c=1}^{C}\Sigma_{c,c}\mu_c^2 + \sum_{c=1}^{C}\Sigma_{c,c}^2\right)$.

Now, we consider the probability of the case $\tilde{x}_{m,n} = \tilde{x}_{i,j} \mid_{m\neq i \vee n\neq j}$ with the above deduced condition, where $i = 1,\ldots,M, j = 1,\ldots,N$ and "$\vee$" denotes logical "OR", which equals to calculate

$$P = \sum_{m=1}^{M}\sum_{n=1}^{N}\left(\sum_{i=1}^{M}\sum_{j=1}^{N}P_{m,n,i,j}\right) - \sum_{m=1}^{M}\sum_{n=1}^{N}P_{m=n\wedge i=j}\,. \tag{6}$$

That is, **as long as** $2\sum_{c=1}^{C}\Sigma_{c,c}\mu_c^2 + \sum_{c=1}^{C}\Sigma_{c,c}^2 \neq 0$ (note that zero-variance will regress $\tilde{\mathrm{x}}$ to be even distributed, instead of Gaussian distributed, which conflicts the realistic condition), we have each

$$P_{m,n,i,j} = P\left(\tilde{x}_{m,n} = \tilde{x}_{i,j} = a_{i,j}\right) = P\left(\tilde{\mathrm{x}} = a_{i,j}\right) = 0\,. \tag{7}$$

That is, we have $P = 0$ for finite times of sampling under the assumed conditions above. Note that $P = 0$ does not mean *impossible*, yet indicates that the chance to meet the discussed condition is *negligible* from the perspective of statistics. Therefore, we can treat IIEU prototype (Equation (4.3)) as a function (instead of a non-function mapping) of $\tilde{x}$ when discussing feature inference.

Appendix .2 Discussion on The Negative Neutralization Effect

By regarding neural activation as a feature re-calibration process, we suppose that a non-important feature possibly deteriorates the updating of the concerned filter if they have an intense negative inner product. This necessitates a selective re-calibration to suppress/emphasize the influence of the meaningless/meaningful features, which clarifies the significance of neural activation. Below we discuss this problem.

.2.1 Discussion

For a quantitative discussion, we consider the following simple settings: for $C \in \mathbb{Z}^+$, let $\boldsymbol{w} \in \mathbb{R}^C, \boldsymbol{w} \neq \mathbf{0}$ be a given vector (*i.e.*, the ideal candidate) and $\mathbf{x} = [\mathrm{x}_c], \mathbf{y} = [\mathrm{y}_c] \in \mathbb{R}^C, c \in \{1,\ldots,C\}$ be two vector-valued random variables (*i.e.*, the alternative candidates); $\mathbf{x},\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$, where $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma}) \mid_{\forall c,\Sigma_{c,c}\neq 0}$ denotes a multivariate normal distribution; $\forall \mathbf{x},\mathbf{y}$, they satisfy the condition $|\langle \boldsymbol{w},\mathbf{x}\rangle| = \kappa_x \leq \kappa_y = |\langle \boldsymbol{w},\mathbf{y}\rangle|, \langle \boldsymbol{w},\mathbf{x}\rangle > 0, \langle \boldsymbol{w},\mathbf{y}\rangle < 0$, where κ_x and κ_y are given (*i.e.*, observed) values. In particular, we use the norm of the expectation $\|\mathbf{x}\| = \sqrt{\langle \mathbf{x},\mathbf{x}\rangle} = \sqrt{\mathbb{E}\left[\mathbf{x}^2\right]}$ to represent the influence of a random variable candidate $\mathbf{x}$ to the given filter $\boldsymbol{w}$ based on Section 4.2.

a. For dimension $C > 1$. As $\boldsymbol{w} \neq \mathbf{0}$, we can find a set of Householder matrices $\{\boldsymbol{H}_c\}$ such that $\boldsymbol{H}_c\boldsymbol{w} = \lambda \boldsymbol{e}_c$, where $\lambda = |\boldsymbol{w}| \in \mathbb{R}^+$. Specifically, the c-th Householder matrix $\boldsymbol{H}_c$ is

computed as:

$$\boldsymbol{H}_c = \boldsymbol{I}_{C\times C} - 2\boldsymbol{h}_c\boldsymbol{h}_c^{\mathrm{T}}\,, \tag{8}$$

where $\boldsymbol{I}_{C\times C}$ is a C-dimensional identity matrix and $\boldsymbol{h}_c$ is the corresponding normal vector of $\boldsymbol{H}_c$ which can be computed as:

$$\boldsymbol{h}_c = \frac{\boldsymbol{w} - |\boldsymbol{w}|\,\boldsymbol{e}_c}{|\boldsymbol{w} - |\boldsymbol{w}|\,\boldsymbol{e}_c|}\,. \tag{9}$$

As such, each $\boldsymbol{H}_c$ is an orthogonal matrix that preserves the norm and inner-product of a random vector, *i.e.*, $\forall \mathbf{x}$, $\|\boldsymbol{H}_c\mathbf{x}\| = \|\mathbf{x}\|$ and $\langle \boldsymbol{H}_c\boldsymbol{w}, \boldsymbol{H}_c\mathbf{x}\rangle = \langle \boldsymbol{w}, \mathbf{x}\rangle$. Then, with the given condition $|\langle \boldsymbol{w}, \mathbf{x}\rangle| = \kappa_x \leq \kappa_y = |\langle \boldsymbol{w}, \mathbf{y}\rangle|\,, \langle \boldsymbol{w}, \mathbf{x}\rangle > 0, \langle \boldsymbol{w}, \mathbf{y}\rangle < 0$, we have:

$$|\langle \boldsymbol{H}_c\boldsymbol{w}, \boldsymbol{H}_c\mathbf{x}\rangle| = |\langle \lambda\boldsymbol{e}_c, \boldsymbol{H}_c\mathbf{x}\rangle| = \lambda\,|(\boldsymbol{H}_c\mathbf{x})_c| = \kappa_x\,, \tag{10}$$

$$|\langle \boldsymbol{H}_c\boldsymbol{w}, \boldsymbol{H}_c\mathbf{y}\rangle| = |\langle \lambda\boldsymbol{e}_c, \boldsymbol{H}_c\mathbf{y}\rangle| = \lambda\,|(\boldsymbol{H}_c\mathbf{y})_c| = \kappa_y\,, \tag{11}$$

$$|(\boldsymbol{H}_c\mathbf{x})_c| = (\boldsymbol{H}_c\mathbf{x})_c = \frac{\kappa_x}{\lambda} \leq \frac{\kappa_y}{\lambda} = |(\boldsymbol{H}_c\mathbf{y})_c| = -\,(\boldsymbol{H}_c\mathbf{y})_c\,. \tag{12}$$

That is, we use $\boldsymbol{H}_c$ to rotate the given filter $\boldsymbol{w}$ to the direction of the base vector $\boldsymbol{e}_c$ such that $\forall \mathbf{x}, \mathbf{y}$, $\boldsymbol{H}_c$ preserves the projections of $\mathbf{x}, \mathbf{y}$ on $\boldsymbol{w}$ after the rotations. As such, we can calculate the conditional expectations of the rotated random vectors by $\mathbb{E}\left[\boldsymbol{H}_c\mathbf{y}\mid_{(\boldsymbol{H}_c\mathbf{y})_c=-\frac{\kappa_y}{\lambda}}\right]$ and $\mathbb{E}\left[\boldsymbol{H}_c\mathbf{x}\mid_{(\boldsymbol{H}_c\mathbf{x})_c=\frac{\kappa_x}{\lambda}}\right]$, respectively. Moreover, as $\boldsymbol{H}_c$ preserves the norms, we have the following corollary for the problem we discuss:

Corollary .2. $\|\mathbf{y}\| \geq \|\mathbf{x}\| \Longleftrightarrow \sqrt{\mathbb{E}\left[(\boldsymbol{H}_c\mathbf{y})^2\mid_{(\boldsymbol{H}_c\mathbf{y})_c=-\frac{\kappa_y}{\lambda}}\right]} \geq \sqrt{\mathbb{E}\left[(\boldsymbol{H}_c\mathbf{x})^2\mid_{(\boldsymbol{H}_c\mathbf{x})_c=\frac{\kappa_x}{\lambda}}\right]}$.

In particular, we first consider $\boldsymbol{H}_c = \boldsymbol{H}_C$ without loss of generality because $\forall i, j$ where $i \neq j$, the swap of the axis-i and -j will not change the norm of a vector. As such, after applying the linear transformations with $\boldsymbol{H}_C$, we have:

$$\boldsymbol{H}_C\mathbf{x}, \boldsymbol{H}_C\mathbf{y} \sim \mathcal{N}\left(\boldsymbol{\mu}', \boldsymbol{\Sigma}'\right)\,, \tag{13}$$

where $\boldsymbol{\mu}' = \boldsymbol{H}_C\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}' = \boldsymbol{H}_c\boldsymbol{\Sigma}\boldsymbol{H}_c^{\mathrm{T}}$. For clarity, following we denote $\boldsymbol{\mu}'$ and $\boldsymbol{\Sigma}'$ as:

$$\boldsymbol{\mu}' = \begin{bmatrix} \boldsymbol{\mu}'_P \\ \mu'_C \end{bmatrix}, \boldsymbol{\Sigma}' = \begin{bmatrix} \boldsymbol{\Sigma}'_{P,P} & \boldsymbol{\Sigma}'_{\boldsymbol{P,C}} \\ \boldsymbol{\Sigma}'_{\boldsymbol{C,P}} & \Sigma'_{C,C} \end{bmatrix}\,, \tag{14}$$

where the index P denotes "from index 1 to $C-1$". Note that $\boldsymbol{\mu}'_P \in \mathbb{R}^{C-1}$ (a column vector), $\mu'_C \in \mathbb{R}$, $\boldsymbol{\Sigma}'_{P,P} \in \mathbb{R}^{C-1\times C-1}$, $\boldsymbol{\Sigma}'_{\boldsymbol{P,C}} \in \mathbb{R}^{C-1}$ (a column vector), $\boldsymbol{\Sigma}'_{\boldsymbol{C,P}} \in \mathbb{R}^{C-1}$ (a row vector), and $\Sigma'_{C,C} \in \mathbb{R}$. Then, with the calculation rules for conditional multivariate

norm distribution, for $\boldsymbol{H}_C\mathbf{y}$, we have:

$$\begin{aligned}
\boldsymbol{\mu}_P^y &= \boldsymbol{\mu}'_P + \boldsymbol{\Sigma}'_{\boldsymbol{P},\boldsymbol{C}} \left(\Sigma'_{C,C}\right)^{-1} \left(-\frac{\kappa_y}{\lambda} - \mu'_C\right) \\
&= \begin{bmatrix} \mu'_1 \\ \mu'_2 \\ \vdots \\ \mu'_{C-1} \end{bmatrix} + \begin{bmatrix} \Sigma'_{1,C} \\ \Sigma'_{2,C} \\ \vdots \\ \Sigma'_{C-1,C} \end{bmatrix} \left(\Sigma'_{C,C}\right)^{-1} \left(-\frac{\kappa_y}{\lambda} - \mu'_C\right) \\
&= \begin{bmatrix} \mu'_1 \\ \mu'_2 \\ \vdots \\ \mu'_{C-1} \end{bmatrix} + \begin{bmatrix} \sigma_1 \left(\kappa'_y - \mu'_C\right) \\ \sigma_2 \left(\kappa'_y - \mu'_C\right) \\ \vdots \\ \sigma_{C-1} \left(\kappa'_y - \mu'_C\right) \end{bmatrix} \\
&= \left[\mu'_c + \sigma'_c \left(\kappa'_y - \mu'_C\right)\right]^{\mathrm{T}}, \qquad (15)
\end{aligned}$$

where $\boldsymbol{\mu}_P^y = \boldsymbol{\mu}'_P \mid_{(\boldsymbol{H}_C\mathbf{y})_C = -\frac{\kappa_y}{\lambda}} \in \mathbb{R}^{C-1}$ denotes the conditional mean vector of $\boldsymbol{\mu}'_P$ with the condition $(\boldsymbol{H}_C\mathbf{y})_C = -\frac{\kappa_y}{\lambda}$; for simplicity, we use σ'_c and κ'_y to denote $\Sigma'_{c,C}\left(\Sigma'_{C,C}\right)^{-1}$ and $-\frac{\kappa_y}{\lambda}$, respectively. Similarly, for $\boldsymbol{H}_C\mathbf{x}$, we have:

$$\begin{aligned}
\boldsymbol{\mu}_P^x &= \boldsymbol{\mu}'_P + \boldsymbol{\Sigma}'_{\boldsymbol{P},\boldsymbol{C}} \left(\Sigma'_{C,C}\right)^{-1} \left(\frac{\kappa_x}{\lambda} - \mu'_C\right) \\
&= \left[\mu'_c + \sigma'_c \left(\kappa'_x - \mu'_C\right)\right]^{\mathrm{T}}, \qquad (16)
\end{aligned}$$

where $\boldsymbol{\mu}_P^x = \boldsymbol{\mu}'_P \mid_{(\boldsymbol{H}_C\mathbf{x})_C = \frac{\kappa_x}{\lambda}} \in \mathbb{R}^{C-1}$ and κ'_x denotes $\frac{\kappa_x}{\lambda}$.

With the above deductions, we have the following deductions for the observed projections κ'_x, κ'_y and conditional mean vectors $\boldsymbol{\mu}_P^x, \boldsymbol{\mu}_P^y$ of the random vector variables $\mathbf{x}, \mathbf{y}$ to ensure the Corollary .2:

$$\begin{aligned}
&\mathbb{E}\left[(\boldsymbol{H}_C\mathbf{y})^2 \mid_{(\boldsymbol{H}_C\mathbf{y})_C = \kappa'_y}\right] = |\boldsymbol{\mu}_P^y|^2 + \left(\kappa'_y\right)^2 \geq |\boldsymbol{\mu}_P^x|^2 + \left(\kappa'_x\right)^2 = \mathbb{E}\left[(\boldsymbol{H}_C\mathbf{x})^2 \mid_{(\boldsymbol{H}_C\mathbf{x})_C = \kappa'_x}\right] \\
\Longrightarrow &\left(\left(\kappa'_y\right)^2 - \left(\kappa'_x\right)^2\right) + \sum_{c=1}^{C-1} \left(\left(\mu'_c + \sigma'_c \left(\kappa'_y - \mu'_C\right)\right)^2 - \left(\mu'_c + \sigma'_c \left(\kappa'_x - \mu'_C\right)\right)^2\right) \geq 0 \\
\Longrightarrow &\left(\left(\kappa'_y\right)^2 - \left(\kappa'_x\right)^2\right) + \sum_{c=1}^{C-1} \sigma'_c \left(\kappa'_y - \kappa'_x\right) \left(2\mu'_c + \sigma'_c \left(\kappa'_y + \kappa'_x - 2\mu'_C\right)\right) \geq 0\,. \qquad (17)
\end{aligned}$$

As $\forall i, j$ where $i \neq j$, the swap of the axis-i and -j does not change the norm of a vector, we can directly replace the axis-C with an axis-c without changing the conclusion. As such, the above deductions can be extended to the general case of $\forall c\colon\ c = 1, 2, \ldots, C$. Based on the above deductions, we identify a simple condition to ensure Corollary .2: $\forall \sigma'_c = 0$, *i.e.*, the transformed covariance matrix $\boldsymbol{\Sigma}'$ is a diagonal matrix such that all of the elements of $\forall \boldsymbol{H}_c\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ are independent. Besides, a particular case is that if the given $\boldsymbol{w}$ and a $\boldsymbol{e}_c$ has the same direction such that it does not require Householder transformations,

then, the Corollary .2 is ensured when $\mathbf{\Sigma}$ is a diagonal matrix (*i.e.*, the elements of $\forall \mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma})$ are independent).

b. For dimension $C = 1$. The condition $C = 1$ ensures $\boldsymbol{w}$ to have the same direction with $\boldsymbol{e}_1$. Then, as $\mathbf{\Sigma} \in \mathbb{R}^{1\times 1}$ is a single-value diagonal matrix, the Corollary .2 is ensured according to the preceding deductions.

Summary. The discussions of the cases **a** and **b** show that the non-important features with intense negative feature-filter inner products possibly neutralize the positive contribution of important features if without selective feature re-calibrations. This clarifies the meaning of neural feature activation.

Appendix .3 Calculations for Section 4.3.2

.3.1 The Range of Term-*S*

In the following, we show the derivations for Equation (4.6) (*i.e.*, the range of the term-*S* of IIEU-B).

We discuss the common case with BN [127] applied (denoted by ψ), *i.e.*, now we have:

$$\tilde{x} \coloneqq \psi\left(\langle \boldsymbol{w}, \boldsymbol{x}\rangle\right) = \gamma \frac{\langle \boldsymbol{w}, \boldsymbol{x}\rangle - \mu}{\sigma} + \beta\,, \tag{18}$$

where $\gamma, \beta \in \mathbb{R}$ denote the channel scaling and shift factors of BN; $\sigma \in \mathbb{R} \neq 0$ and $\mu \in \mathbb{R}$ denote the standard deviation and mean of $\tilde{x}$ for the channel-c (*i.e.*, the current channel).

Let $E = \|\boldsymbol{x}\|\,\|\boldsymbol{w}\| \neq 0$. As the vanilla cosine similarity $\frac{\langle \boldsymbol{w}, \boldsymbol{x}\rangle}{\|\boldsymbol{w}\|\|\boldsymbol{x}\|} \in [0, 1]$, the codomain of term-*S* , *i.e.*, $\frac{\tilde{x}}{\|\boldsymbol{w}\|\|\boldsymbol{x}\|}$ can be calculated as:

$$\begin{cases} -\frac{|\gamma|}{\sigma} - \frac{|\gamma|\mu}{E\sigma} + \frac{\beta}{E} \leq \frac{\tilde{x}}{E} \leq \frac{|\gamma|}{\sigma} - \frac{|\gamma|\mu}{E\sigma} + \frac{\beta}{E}\,, & \gamma \geq 0\,, \\ -\frac{|\gamma|}{\sigma} + \frac{|\gamma|\mu}{E\sigma} + \frac{\beta}{E} \leq \frac{\tilde{x}}{E} \leq \frac{|\gamma|}{\sigma} + \frac{|\gamma|\mu}{E\sigma} + \frac{\beta}{E}\,. & \gamma < 0\,. \end{cases} \tag{19}$$

Then, let $r = \frac{\gamma}{\sigma}$, we have:

$$-|r| + \frac{\beta - r\mu}{E} \leq \frac{\tilde{x}}{E} \leq |r| + \frac{\beta - r\mu}{E}\,, \tag{20}$$

i.e., the Equation (4.6).

.3.2 The Derivative of Term-*S* about $\boldsymbol{w}$

In this Appendix, we show the calculational details of Equation (4.7), *i.e.*, the (partial) derivative of the term-*S* $s(\boldsymbol{w})$ about $\boldsymbol{w}$ ($\nabla_{\boldsymbol{w}} s(\boldsymbol{w})$) as follows:

$$\begin{aligned}\nabla_{\boldsymbol{w}} s(\boldsymbol{w}) = \nabla_{\boldsymbol{w}} \frac{\langle \boldsymbol{w}, \boldsymbol{x} \rangle}{\|\boldsymbol{w}\| \|\boldsymbol{x}\|} &= \|\boldsymbol{x}\|^{-1} \left(\frac{\partial \|\boldsymbol{w}\|^{-1}}{\partial \boldsymbol{w}} \cdot \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{x} \cdot \|\boldsymbol{w}\|^{-1} \right) \\ &= \|\boldsymbol{x}\|^{-1} \left(- \|\boldsymbol{w}\|^{-2} \cdot \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|} \cdot \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + \frac{\boldsymbol{x}}{\|\boldsymbol{w}\|} \right) \\ &= \|\boldsymbol{x}\|^{-1} \left(\frac{\|\boldsymbol{w}\|^2 \boldsymbol{x} - \boldsymbol{w}\boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}}{\|\boldsymbol{w}\|^3} \right) \\ &= \frac{\|\boldsymbol{w}\|^2 \boldsymbol{x} - \boldsymbol{w}\boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}}{\|\boldsymbol{x}\| \|\boldsymbol{w}\|^3} . \end{aligned} \tag{21}$$

.3.3 The Derivative of Term-*B* about $\boldsymbol{w}$

In this Appendix, we show the calculational details of Equation (4.8), *i.e.*, the (partial) derivative of the term-*B* $\nu(\boldsymbol{w})$ about $\boldsymbol{w}$ ($\nabla_{\boldsymbol{w}} \nu(\boldsymbol{w})$) as follows:

$$\begin{aligned}\nabla_{\boldsymbol{w}} \nu(\boldsymbol{w}) = \nabla_{\boldsymbol{w}} \delta\left(\dot{\gamma} \overline{\langle \boldsymbol{w}, \boldsymbol{x} \rangle} + \dot{\beta} \right) &= \frac{\partial \delta\left(\dot{\gamma} \overline{\langle \boldsymbol{w}, \boldsymbol{x} \rangle} + \dot{\beta} \right)}{\partial \left(\dot{\gamma} \overline{\langle \boldsymbol{w}, \boldsymbol{x} \rangle} + \dot{\beta} \right)} \cdot \frac{\partial \left(\dot{\gamma} \overline{\langle \boldsymbol{w}, \boldsymbol{x} \rangle} + \dot{\beta} \right)}{\partial \boldsymbol{w}} \\ &= \delta\left(\dot{\gamma} \overline{\langle \boldsymbol{w}, \boldsymbol{x} \rangle} + \dot{\beta} \right) \left(1 - \delta\left(\dot{\gamma} \overline{\langle \boldsymbol{w}, \boldsymbol{x} \rangle} + \dot{\beta} \right) \right) \cdot \dot{\gamma} \cdot \frac{1}{N} \cdot \sum\nolimits_{n=1}^{N} \boldsymbol{x}(n) \\ &= \delta\left(\frac{\dot{\gamma}}{N} \boldsymbol{w}^{\mathrm{T}} \sum\nolimits_{n=1}^{N} \boldsymbol{x}(n) \right) \left(1 - \delta\left(\frac{\dot{\gamma}}{N} \boldsymbol{w}^{\mathrm{T}} \sum\nolimits_{n=1}^{N} \boldsymbol{x}(n) \right) \right) \cdot \frac{\dot{\gamma}}{N} \sum\nolimits_{n=1}^{N} \boldsymbol{x}(n) \\ &= \delta\left(\dot{\gamma} \boldsymbol{w}^{\mathrm{T}} \overline{\boldsymbol{x}} + \dot{\beta} \right) \left(1 - \delta\left(\dot{\gamma} \boldsymbol{w}^{\mathrm{T}} \overline{\boldsymbol{x}} + \dot{\beta} \right) \right) \dot{\gamma} \overline{\boldsymbol{x}} , \end{aligned} \tag{22}$$

where $N = H \times L$ denotes the number of feature vectors in the current feature map (a tensor) of the layer-τ (*i.e.*, $\mathbf{X}$ with a spatial resolution of $H \times L$, as assumed in Section 4.2 (Preliminaries)). Note that δ denotes the Sigmoid function and we adopt the known derivation rule of the Sigmoid function, *i.e.*, $\forall x \in \mathbb{R}$, $\delta(x) = \delta(x)(1 - \delta(x))$. This derivation rule can be directly generalized to the case of vector-valued inputs.

.3.4 Calculation of Equation (4.9)

From Equation (4.9), we identify term-*S* enabling each neuron to model detailed cross-channel feature-filter interactions at every spatial coordinate and leverage these informative cues to improve the filter updating. In the following, we show the calculational details of

Equation (4.9):

$$\begin{aligned}\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} &= \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \\ \vdots \\ \boldsymbol{w}_C \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_1 & \boldsymbol{w}_2 & \dots & \boldsymbol{w}_C \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_C \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{w}_1\boldsymbol{w}_1 & \boldsymbol{w}_1\boldsymbol{w}_2 & \dots & \boldsymbol{w}_1\boldsymbol{w}_C \\ \boldsymbol{w}_2\boldsymbol{w}_1 & \boldsymbol{w}_2\boldsymbol{w}_2 & \dots & \boldsymbol{w}_2\boldsymbol{w}_C \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{w}_C\boldsymbol{w}_1 & \boldsymbol{w}_C\boldsymbol{w}_2 & \dots & \boldsymbol{w}_C\boldsymbol{w}_C \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_C \end{bmatrix} \\ &= \boldsymbol{w}\left(\sum\nolimits_{c=1}^{C} \boldsymbol{w}_c\boldsymbol{x}_c\right) = \left(\sum\nolimits_{c=1}^{C} \boldsymbol{w}_c\boldsymbol{x}_c\right)\boldsymbol{w}\,. \end{aligned} \tag{23}$$

.3.5 Proof of The Inequality 4.14: $|\nabla_{\boldsymbol{w}}\nu(\boldsymbol{w})| \leq \frac{1}{4}|\dot{\gamma}|\,|\overline{\boldsymbol{x}}|$

First, we adopt the conclusion for $\nabla_{\boldsymbol{w}}\nu(\boldsymbol{w})$ in Section .3.3:

$$\nabla_{\boldsymbol{w}}\nu(\boldsymbol{w}) = \delta\left(\dot{\gamma}\boldsymbol{w}^{\mathrm{T}}\overline{\boldsymbol{x}} + \dot{\beta}\right)\left(1 - \delta\left(\dot{\gamma}\boldsymbol{w}^{\mathrm{T}}\overline{\boldsymbol{x}} + \dot{\beta}\right)\right)\dot{\gamma}\overline{\boldsymbol{x}}\,. \tag{24}$$

As $\dot{\gamma}, \dot{\beta} \in \mathbb{R}$, $\boldsymbol{w}, \overline{\boldsymbol{x}} \in \mathbb{R}^C$, and $\boldsymbol{w}^{\mathrm{T}}\overline{\boldsymbol{x}} \in \mathbb{R}$, let $z = \dot{\gamma}\boldsymbol{w}^{\mathrm{T}}\overline{\boldsymbol{x}} + \dot{\beta} \in \mathbb{R}$ without loss of generality. Then, we have:

$$\begin{aligned}|\nabla_{\boldsymbol{w}}\nu(\boldsymbol{w})| &= \left|\delta\left(\dot{\gamma}\boldsymbol{w}^{\mathrm{T}}\overline{x} + \dot{\beta}\right)\left(1 - \delta\left(\dot{\gamma}\boldsymbol{w}^{\mathrm{T}}\overline{\boldsymbol{x}} + \dot{\beta}\right)\right)\dot{\gamma}\overline{\boldsymbol{x}}\right| = \left|\delta(z)\left(1-\delta(z)\right)\dot{\gamma}\overline{\boldsymbol{x}}\right| \\ &\leq \sup\left(\left|\delta(z)\left(1-\delta(z)\right)\right|\right)\cdot|\dot{\gamma}|\cdot|\overline{\boldsymbol{x}}| = \frac{1}{2}\left(1-\frac{1}{2}\right)|\dot{\gamma}|\,|\overline{\boldsymbol{x}}| = \frac{1}{4}|\dot{\gamma}|\,|\overline{\boldsymbol{x}}|\,. \end{aligned} \tag{25}$$

That is: $|\nabla_{\boldsymbol{w}}\nu(\boldsymbol{w})| \leq \frac{1}{4}|\dot{\gamma}|\,|\overline{\boldsymbol{x}}|$. Therefore, we complete the proof. ■

Appendix .4 Discussion of AdaShift: from MCDM-inspired Intuitions and Properties

In this Appendix, we discuss the attributes of AdaShift-B (as the representative of the practical AdaShift family) in light of the basic intuitions and assumed properties of neural activation inspired by our MCDM hypothesis (introduced in Section 4.3). Below we demonstrate that AdaShift-B is consistent with the basic MCDM-inspired intuitions and holds the corresponding (assumed) properties.

Property 4.1 (The Directional Monotonicity $\wedge$ Sign Constraint of $\varsigma(\varrho_x)$ About ϱ_x). In Section 4.3.1, we introduce the physical (intuitive) meaning and the definition of Property 4.1 based on Intuition 4.2. In particular, Property 4.1 can be ensured by the assumed conditions of Proposition 4.1 (as detailed in Appendix .1.1). Following we show that AdaShift-B satisfies these assumed conditions, *i.e.*, ***AdaShift-B holds Property 4.1***.

Discussion. First, AdaShift-B, *i.e.*, $\phi(\tilde{x}) = \rho(\tilde{x})\,\tilde{x} = \varsigma(\tilde{x}+\Delta)\,\tilde{x}$, learns to approximate the ideal similarity measure $\varrho(\tilde{x})$ of the input $\tilde{x}$ by $\hat{\varrho}_x = \hat{\varrho}(\tilde{x}) = \tilde{x}+\Delta$, where the adaptive shift factor Δ is defined by Equation (5.8) and the adjuster ς is an vanilla Sigmoid function of $\hat{\varrho}_x$, *i.e.*, $\varsigma(\hat{\varrho}_x) = \frac{\hat{\varrho}_x}{1+\hat{\varrho}_x}$. Therefore, $\forall \hat{\varrho}_x \in \mathbb{R}$, $\varsigma(\hat{\varrho}_x)$ is monotonically non-decreasing about $\hat{\varrho}_x \wedge \varsigma(\hat{\varrho}_x) \geq 0$. That is, AdaShift-B (with a Sigmoid adjuster) naturally satisfies the assumed conditions of Proposition 4.1. ***This ensures Proposition 4.1 for AdaShift-B***.

Property 4.2 (CNI).
In Section 4.3.1, we introduce the physical (intuitive) meaning and the definition of Property 4.2 (CNI) based on Intuition 4.4. In particular, the strict case (*i.e.*, case .1) of Property 4.2 can be ensured by the assumed conditions of Proposition .1 (as detailed in Appendix Sec. .1.2). Following we show that AdaShift-B satisfies these assumed conditions, *i.e.*, ***AdaShift-B holds Property 4.2***.

Discussion. The core spirit of Property 4.2 is to selectively constrain the influence of non-important (negative) alternative candidates (*i.e.*, features). We first analyze a simple case and then generalize it to an extended case.

Simple case. In the simple case, we take into account the re-scaling effect of the Z-Scoring of LayerNorm. That is, Δ (*i.e.*, Δ_c, which we specified as in the subsequent text for clarity) can be expressed by:

$$\Delta_c = \left[\mathrm{LN}\left(\mathrm{avgpool}_{H\times L}\left(\tilde{\mathbf{X}}\right)\right)\right]_c = \mathrm{LN}\left(\bar{\tilde{x}}_c\right) = \dot{\gamma}_c \frac{\bar{\tilde{x}}_c - \mu_{\bar{x}}}{\delta_{\bar{x}}} + \dot{\beta}_c\,, \tag{26}$$

where $\mu_{\bar{x}}$ and $\delta_{\bar{x}}$ denote the mean value and (unbiased estimation of) standard deviation of channel mean vector (*i.e.*, $\bar{\tilde{\boldsymbol{x}}}$, where $\bar{\tilde{x}}_c$ is the c-th element of $\bar{\tilde{\boldsymbol{x}}}$), respectively. $\dot{\gamma}_c$ and $\dot{\beta}_c$ denote the channel-wise scaling and shift factors (applied to introduce the parametric element-wise affine of LayerNorm), respectively.

Then, for $\hat{\varrho}_x \to -\infty$ (*i.e.*, $\tilde{x}_c + \Delta_c - \infty$), only two cases are possible, *i.e.*, (1) $\tilde{x}_c \to -\infty$; (2) $\tilde{x}_c \to +\infty$. Note that here we solely discuss the function of the concerned input, *i.e.*, $\tilde{x}_c$, so other terms (*e.g.*, $\tilde{x}_i, i \neq c$) are treated as known values.

Below we first consider the case (1) and generalize the deduced conclusion to the case (2). For case (1), we have:

$$\lim_{\tilde{x}_c\to-\infty} \mu_{\bar{x}} = \lim_{\tilde{x}_c\to-\infty} \bar{\tilde{x}}_c - \frac{1}{C}\bar{\tilde{x}}_c - \frac{1}{C}\sum_{i=1,i\neq c}^{C} \bar{\tilde{x}}_i = \frac{C-1}{C}\bar{\tilde{x}}_c - \frac{1}{C}\sum_{i=1,i\neq c}^{C} \bar{\tilde{x}}_i\,, \tag{27}$$

so that we only need to consider the term $\frac{C-1}{C}\bar{\tilde{x}}_c$ and denote this term as $\kappa_\mu \tilde{x}_c$, where $\lim_{\tilde{x}_c\to-\infty} \kappa_\mu = \frac{C-1}{C\cdot N}$, where $N = H \cdot L$ denotes the total number of pixel locations of the current feature map. Therefore, κ_μ is a finite value.

Similarly, for $\delta_{\bar{x}}$, we have:

$$\begin{aligned}\lim_{\tilde{x}_c\to-\infty}\delta_{\bar{x}} &= \lim_{\tilde{x}_c\to-\infty}\sqrt{\frac{\sum_{i=1}^{C}\left(\bar{\tilde{x}}_i-\mu_{\bar{x}}\right)^2}{C-1}} = \lim_{\tilde{x}_c\to-\infty}\frac{\sqrt{\left(\bar{\tilde{x}}_c-\bar{\tilde{x}}_c\right)+\sum_{i=1,i\neq c}^{C}\left(\bar{\tilde{x}}_i-\mu_{\bar{x}}\right)^2}}{\sqrt{C-1}}\\ &= \lim_{\tilde{x}_c\to-\infty}\frac{\sqrt{\sum_{i=1,i\neq c}^{C}\left(\bar{\tilde{x}}_i-\mu_{\bar{x}}\right)^2}}{\sqrt{C-1}} = \lim_{\tilde{x}_c\to-\infty}\frac{\sqrt{\sum_{i=1,i\neq c}^{C}\left(\left(\kappa_\mu\tilde{x}_c\right)^2-2\bar{\tilde{x}}_i\left(\kappa_\mu\tilde{x}_c\right)+\bar{\tilde{x}}_i^2\right)}}{\sqrt{C-1}}\\ &\to \frac{1}{\sqrt{C-1}}\sqrt{\sum_{i=1,i\neq c}^{C}\left(\left(\kappa_\mu\tilde{x}_c\right)^2-2\bar{\tilde{x}}_i\left(\kappa_\mu\tilde{x}_c\right)\right)}. \end{aligned} \tag{28}$$

Note that as $\delta_{\bar{x}}$ is also essentially a first-order value of $\tilde{x}_c$ (like $\mu_{\bar{x}}$) and only $\left(\kappa_\mu\tilde{x}_c\right)^2$ is the first-order value of $\tilde{x}_c$ in $\delta_{\bar{x}}$, we can also represent it by $\kappa_\sigma\tilde{x}_c$, where

$$\begin{aligned}\lim_{\tilde{x}_c\to-\infty}|\kappa_\sigma| &= \lim_{\tilde{x}_c\to-\infty}\frac{\frac{1}{\sqrt{C-1}}\sqrt{\left(\bar{\tilde{x}}_c-\mu_{\bar{x}}\right)^2+\sum_{i=1,i\neq c}^{C}\left(\left(\kappa_\mu\tilde{x}_c\right)^2-2\bar{\tilde{x}}_i\left(\kappa_\mu\tilde{x}_c\right)+\bar{\tilde{x}}_i^2\right)}}{|\tilde{x}_c|}\\ &= \lim_{\tilde{x}_c\to-\infty}\frac{\frac{1}{\sqrt{C-1}}\sqrt{\left(\bar{\tilde{x}}_c-\mu_{\bar{x}}\right)^2+\sum_{i=1,i\neq c}^{C}\left(\kappa_\mu\tilde{x}_c\right)^2}}{|\tilde{x}_c|}\\ &= \lim_{\tilde{x}_c\to-\infty}\frac{\frac{1}{\sqrt{C-1}}\sqrt{\left(\bar{\tilde{x}}_c-\mu_{\bar{x}}\right)^2+(C-1)\left(\kappa_\mu\tilde{x}_c\right)^2}}{|\tilde{x}_c|}\\ &= \lim_{\tilde{x}_c\to-\infty}\frac{1}{\sqrt{C-1}}\sqrt{\frac{\left(\bar{\tilde{x}}_c-\mu_{\bar{x}}\right)^2+(C-1)\left(\kappa_\mu\tilde{x}_c\right)^2}{\tilde{x}_c^2}}\\ &= \lim_{\tilde{x}_c\to-\infty}\frac{1}{\sqrt{C-1}}\sqrt{\left(\frac{1}{N}-\kappa_\mu\right)^2+(C-1)\left(\kappa_\mu\right)^2}\\ &= \sqrt{\frac{\left(\frac{1}{N}-\kappa_\mu\right)^2}{C-1}+\kappa_\mu^2}. \end{aligned} \tag{29}$$

That is, we can re-write Δ_c when $\tilde{x}_c\to-\infty$ as:

$$\lim_{\tilde{x}_c\to-\infty}\Delta_c = \dot{\gamma}_c\frac{\frac{1}{N}\tilde{x}_c-\kappa_\mu\tilde{x}_c}{-\kappa_\sigma\tilde{x}_c}+\dot{\beta}_c = \dot{\gamma}_c\frac{\frac{1}{N}\tilde{x}_c-\kappa_\mu\tilde{x}_c}{-\kappa_\sigma\tilde{x}_c}+\dot{\beta}_c = -\dot{\gamma}_c\frac{\frac{1}{N}-\kappa_\mu}{\kappa_\sigma}+\dot{\beta}_c, \tag{30}$$

which regresses to a known value that consists of N, κ_μ, κ_σ, $\dot{\gamma}_c$, ***and*** $\dot{\beta}_c$.

With the above deduced conclusion, we have:

$$\begin{aligned}\lim_{\hat{\varrho}(\tilde{x})\to-\infty}\left(\varsigma\left(\hat{\varrho}\left(\tilde{x}\right)\right)\tilde{x}\right) &= \lim_{(\tilde{x}+\Delta)\to-\infty}\operatorname{sigmoid}\left(\tilde{x}+\Delta\right)\tilde{x} = \lim_{(\tilde{x}+\Delta)\to-\infty}\frac{e^{\tilde{x}+\Delta}}{e^{\tilde{x}+\Delta}+1}\cdot\tilde{x}\\ &= \lim_{z\to+\infty}\frac{e^{-z}}{e^{-z}+1}\cdot\left(-z-\Delta\right)|_{z=-(\tilde{x}+\Delta)} = \lim_{z\to+\infty}\frac{-\frac{z+\Delta}{e^{z}}}{e^{-z}+1}\\ &= \frac{0}{0+1}=0\,. \end{aligned} \tag{31}$$

This conclusion can be simply generalized to the case (2) (*i.e.*, $\tilde{x}_c\to+\infty$).

Therefore, AdaShift-B holds (the strict case of) Property 4.2.

Extended case. For more generality, we discuss an extended case where

$$\Delta_c = \dot{\gamma}_c\bar{\tilde{x}}_c + \dot{\beta}_c\,. \tag{32}$$

That is, the Z-Scoring which regresses Δ_c to a known value is removed to relax the value constraint of Δ_c.

With the above condition, similar to the simple case, $\hat{\varrho}\left(\tilde{x}\right)\to-\infty$ is only possible for (1) $\tilde{x}_c\to-\infty$ and (2) $\tilde{x}_c\to+\infty$.

Then, for $\tilde{x}_c\to-\infty$, we have:

$$\begin{aligned}\lim_{\hat{\varrho}(\tilde{x})\to-\infty}\left(\varsigma\left(\hat{\varrho}\left(\tilde{x}\right)\right)\tilde{x}\right) &= \lim_{(\tilde{x}+\Delta)\to-\infty}\operatorname{sigmoid}\left(\tilde{x}+\Delta\right)\tilde{x}\\ &= \lim_{(\tilde{x}+\Delta)\to-\infty}\frac{e^{\tilde{x}+\Delta}}{e^{\tilde{x}+\Delta}+1}\tilde{x}\\ &= \lim_{\left(\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c\right)\to-\infty}\frac{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}}{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}+1}\cdot\tilde{x}\\ &= \lim_{z\to+\infty}\frac{e^{-z}}{e^{-z}+1}\cdot-\frac{z+\dot{\beta}_c}{1+\frac{\dot{\gamma}_c}{N}}\;|_{z=-\left(\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c\right)}\\ &= \lim_{z\to+\infty}\left(\frac{-N}{N+\dot{\gamma}_c}\cdot\frac{\frac{z+\dot{\beta}_c}{e^{z}}}{e^{-z}+1}\right)\\ &= \frac{-N}{N+\dot{\gamma}_c}\cdot\frac{0}{0+1}\\ &= 0\,. \end{aligned} \tag{33}$$

This ensure that the (the strict case of) Property 4.2 holds for the extended case (of AdaShift-B).

Property 4.3 (PPI).

In Section 4.3.1, we introduce the physical (intuitive) meaning and the definition of Property 4.3 (PPI) based on Intuition 4.5. Following we show that AdaShift-B satisfies these assumed conditions, *i.e.*, ***AdaShift-B holds Property 4.3***.

Discussion. The core spirit of Property 4.3 is to preserve the positive influence of different

important alternative candidates (*i.e.*, features). We analyze this based on the simple case and the extended case of Δ_c introduced in the above discussion of Property 4.2.

In particular, Property 4.3 can be satisfied by the *Lipschitz continuity* of the overall activation function $\phi(\tilde{x}_c)$ according to its definition. Therefore, below we confirm the Lipschitz continuity of $\phi(\tilde{x}_c)$ of AdaShift-B.

First, the (partial) derivative of AdaShift-B about the concerned input $\tilde{x}_c$ can be calculated by:

$$\begin{aligned}\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right) &= \frac{\partial\left(\text{sigmoid}\left(\tilde{x}_c+\Delta\right)\cdot\tilde{x}_c\right)}{\partial\tilde{x}_c} \\ &= \frac{\partial\text{sigmoid}\left(\tilde{x}_c+\Delta\right)}{\partial\tilde{x}_c}\cdot\frac{\partial\left(\tilde{x}_c+\Delta\right)}{\partial\tilde{x}_c}\cdot\tilde{x}_c+\text{sigmoid}\left(\tilde{x}_c+\Delta\right). \end{aligned} \tag{34}$$

For the simple case where Z-Scoring is applied, Δ_c is always a bounded value that has weak relationship to the input $\tilde{x}_c$ (*i.e.*, can be regarded as a zero-order value of $\tilde{x}_c$). Therefore, we can treat Δ_c as a known value in the calculation of $\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right)$, *i.e.*,

$$\begin{aligned}\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right) &= \frac{\partial\text{sigmoid}\left(\tilde{x}_c+\Delta\right)}{\partial\tilde{x}_c}\cdot\tilde{x}_c+\text{sigmoid}\left(\tilde{x}_c+\Delta\right) \\ &= \frac{e^{\tilde{x}+\Delta}}{e^{\tilde{x}+\Delta}+1}\cdot\left(1-\frac{e^{\tilde{x}+\Delta}}{e^{\tilde{x}+\Delta}+1}\right)\cdot\tilde{x}_c+\text{sigmoid}\left(\tilde{x}_c+\Delta\right) \\ &= \frac{\tilde{x}_c\cdot e^{\tilde{x}+\Delta}}{\left(e^{\tilde{x}+\Delta}+1\right)^2}+\text{sigmoid}\left(\tilde{x}_c+\Delta\right) \\ &< \frac{\tilde{x}_c\cdot e^{\tilde{x}+\Delta}}{\left(e^{\tilde{x}+\Delta}+1\right)^2}+1\,. \end{aligned} \tag{35}$$

So, the boundedness of $\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right)$ is only possibly violated when $\tilde{x}_c\to+\infty$. But, as

$$\begin{aligned}\lim_{\tilde{x}\to+\infty}\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right) &= \lim_{\tilde{x}\to+\infty}\left(\frac{\tilde{x}_c\cdot e^{\tilde{x}+\Delta}}{\left(e^{\tilde{x}+\Delta}+1\right)^2}+1\right)=\lim_{\tilde{x}\to+\infty}\left(\frac{\tilde{x}_c}{e^{\tilde{x}+\Delta}+1}\cdot\frac{e^{\tilde{x}+\Delta}}{e^{\tilde{x}+\Delta}+1}+1\right) \\ &= 0\cdot 1+1=1\,. \end{aligned} \tag{36}$$

Therefore, $\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right)$ is bounded on the whole domain, *i.e.*, $\phi(\tilde{x}_c)$ of AdaShift-B holds Lipschitz continuity. ***This ensures that Property 4.3 holds for AdaShift-B***.

Similarly, for the extended case, we have:

$$\begin{aligned}
\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right) &= \frac{\partial \text{sigmoid}\left(\tilde{x}_c+\Delta\right)}{\partial \tilde{x}_c}\cdot\frac{\partial\left(\tilde{x}_c+\Delta\right)}{\partial \tilde{x}_c}\cdot\tilde{x}_c+\text{sigmoid}\left(\tilde{x}_c+\Delta\right)\\
&< \frac{\partial \text{sigmoid}\left(\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c\right)}{\partial \tilde{x}_c}\cdot\left(1+\frac{\dot{\gamma}_c}{N}\right)\cdot\tilde{x}_c+1\\
&= \frac{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}}{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}+1}\cdot\left(1-\frac{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}}{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}+1}\right)\cdot\left(1+\frac{\dot{\gamma}_c}{N}\right)\cdot\tilde{x}_c+1\\
&= \frac{\left(1+\frac{\dot{\gamma}_c}{N}\right)\cdot\tilde{x}_c}{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}+1}\cdot\frac{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}}{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}+1}+1\,.
\end{aligned}\tag{37}$$

Note that $|\dot{\gamma}_c|$ is a small value in common due to the $\mathcal{L}_2$ regularization applied on learnable parameters and N is typically a relatively big value (*i.e.*, $|\dot{\gamma}_c| \ll N$). That is, we have:

$$\begin{aligned}
\lim_{\tilde{x}\to+\infty}\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right) &= \lim_{\tilde{x}\to+\infty}\left(\frac{\partial \text{sigmoid}\left(\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c\right)}{\partial \tilde{x}_c}\cdot\left(1+\frac{\dot{\gamma}_c}{N}\right)\cdot\tilde{x}_c+1\right)\\
&= \lim_{\tilde{x}\to+\infty}\left(\frac{\left(1+\frac{\dot{\gamma}_c}{N}\right)\cdot\tilde{x}_c}{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}+1}\cdot\frac{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}}{e^{\left(1+\frac{\dot{\gamma}_c}{N}\right)\tilde{x}_c+\dot{\beta}_c}+1}+1\right)\\
&= 0\cdot 1+1\\
&= 1\,.
\end{aligned}\tag{38}$$

Therefore, $\nabla_{\tilde{x}_c}\phi\left(\tilde{x}_c\right)$ is also bounded, which ensures the Lipschitz continuity of $\phi(\tilde{x}_c)$ of AdaShift-B for the extended case. ***This generalizes Property 4.3 of AdaShift-B for the extended case***.

Property 4.4 (OD).
In Section 4.3.1, we introduce the physical (intuitive) meaning and the definition of Property 4.4 (OD) based on Intuition 4.6.

Discussion. The core spirit of Property 4.4 is to prevent the gradient and feature vanishing after neural activation. It expects the space of the difference of the upper-bound and lower-bound of a re-weighting function to be adequate.

Note that Property 4.4 is a relaxed constraint since the trainable parameters of a (recent) neural network layer (*e.g.*, linear layer and parametric normalization layer) are capable of providing an extent of flexibility to the (intensities of) features.

For AdaShift-B, where $\rho\left(\tilde{x}\right) = \text{sigmoid}\left(\tilde{x}+\Delta\right) \in (0,1)$, it is easy to satisfy the condition of Property 4.4, as

$$\lim_{(\tilde{x}+\Delta)\to+\infty}\varsigma\left(\tilde{x}+\Delta\right)-\lim_{(\tilde{x}+\Delta)\to-\infty}\varsigma\left(\tilde{x}+\Delta\right)=1-0=1\,.\tag{39}$$

This shows that AdaShift-B holds Property 4.4.

Appendix .5 Qualitative Discussion on AdaShift and IIEU from MCDM Hypothesis

In Section 5.5.1, we demonstrate our MCDM-induced neural activation models, *i.e.*, the practical AdaShifts and IIEUs both outperform past SOTA activation models by a significant margin while they exhibit notable differences in attributes. Specifically, using AdaShift-B and IIEU-B (Section 4.3.2) as representative examples of their kinds, we find IIEU-B enhances small-size backbones (*e.g.*, ResNet-14 and -26 [11]) more remarkably, and in contrast, AdaShift-B shows further improvements on deeper backbones (*e.g.*, ResNet-50 and -101 [11]) with higher practical efficiency. Their qualitative differences are actually interpretable from our MCDM hypothesis. Furthermore, we think these experimental phenomena reveal meaningful findings and their MCDM-based interpretations can be helpful for the choice and design of neural activation models.

That is, despite that AdaShift and IIEU both hold the basic properties of neural activation inspired by MCDM hypothesis, their approaches to address the *mismatched feature scoring* problem (inferred from the MCDM hypothesis) are different and these introduce notable differences in the mechanism.

IIEU solves mismatched feature scoring through a direct norm-decoupling strategy for features and filters, which eliminates the possible norm-based biases that take away from the features' actual (*i.e.*, unbiased) importance. This solution is straightforward and targeted.

In contrast, AdaShift introduces relatively gentle adjustments by leveraging different ranges of local and non-local cues jointly and adaptively to improve the re-weighting process of self-gated input re-calibration. We realize the key of this idea by rethinking the meaning of feature and filter (2-)norms in a typical Softmax-based classification process, where the physical meaning of features and filters are clear, and then generalize the understandings to common learning layers. This solution is soft, which leaves more room for adaptive parametric adjustments.

So, for *mismatched feature scoring* problem, from the confidence of classification/recognition, compared to a deeper/stronger neural network backbone, a smaller/weaker backbone likely learns relatively weaker representations that have higher uncertainty of recognition. As analyzed in Section 5.4.1, we identify that feature and filter norms can be influential in recognition. Yet, especially, *the influence of filters and filter norms on recognition in inference is seldom discussed in the past works*. That is, w.r.t. the learning in a neuron, **a weaker neural network not only (likely) generates unreliable features/feature norms but also learns relatively unreliable filters/filter norms that serve as the ideal candidates for information selection and feature generation**. Therefore, the direct norm-decoupling casted by IIEU, which provides straightforward and targeted rectifications to feature-filter similarities, brings more remarkable improvements on smaller backbones. A deeper network, in contrast, is expected to produce relatively reliable features/feature norms with the learned comparatively reliable filters/filter norms, so that it turns out to be more suitable for gentle

and adaptive adjustments. This explains why AdaShift introduces more improvements on deeper backbones than IIEU.

Appendix .6 Qualitative Assessment of Activation Model Based on MCDM Hypothesis

In this Appendix, we show the significant potential of our MCDM hypothesis for interpreting the working mechanism of neural activation. To the best of our knowledge, our MCDM hypothesis is the first and only specialized hypothesis capable of providing predictive qualitative assessments of neural activation models. Although two general principles, *i.e.*, *nonlinearity* and *Lipschitz continuity (but not always)* have been adopted to analyze neural activation functions, **before MCDM hypothesis was proposed, the field of *effective practical instructions/tutorials for designing neural activation models (functions/non-function mappings) from scratch*, however, was still unexplored**.

By leveraging our MCDM hypothesis, we provide predictive qualitative preliminary assessments for representative neural activation models of different types based on the new intuitions and properties (introduced in Section 4.3 and Section 5.4.1). The effectiveness of the qualitative assessment is then experimentally validated on CIFAR-100 dataset [28] with CIFAR-ResNet-56 backbone [11] (the public version [29] is adopted).

For the above purposes, we introduce 3 different targeted control groups using **Sigmoid, Tanh, and ERF as the base functions**, respectively. Based on each base function, we then suggest **a series of modified activation functions to validate the proposed properties of neural activation inspired by the MCDM hypothesis**. Note that we first let the approximated similarity $\hat{\varrho}_x = \hat{\varrho}(\tilde{x}) = \tilde{x}$, *i.e.*, the common condition applied for past works, to evaluate **the 4 basic properties, *i.e.*, Property 4.1 (the directional monotonicity $\wedge$ sign constraint of $\varsigma(\varrho_x)$ about ϱ_x), Property 4.2 (CNI), Property 4.3 (PPI), and Property 4.4 (OD)**, and then consider $\hat{\varrho}(\tilde{x}) = \tilde{x}+\Delta$, *i.e.*, **the corresponding modified AdaShift-B(s) to validate the effectiveness of the intuition of *mismatched feature scoring* (Intuition 4.3)**.

Below we introduce the standardized forms of the series functions of different control groups for the evaluations of the corresponding properties. Specifically, we first specify the base forms as preliminaries and introduce their self-gated forms that satisfy all the assumed basic properties as the original self-gated functions. We then modify the self-gated forms based on the corresponding MCDM-inspired properties with a controlled variable method (*i.e.*, to violate or to meet the concerned property intentionally, by giving tailored modifications). In particular, **we use ReLU function as the global baseline of these evaluations**. Note that although rigid and simple, ReLU holds the 4 basic properties of MCDM hypothesis (with the condition $\hat{\varrho}_x = \tilde{x}$). This makes ReLU a suitable baseline for our discussion.

Base form.

$$\phi_i^{(0)}(\tilde{x}) = \varsigma_i^{(0)}(\tilde{x})\,\tilde{x}, i = 1, 2, 3\,, \tag{40}$$

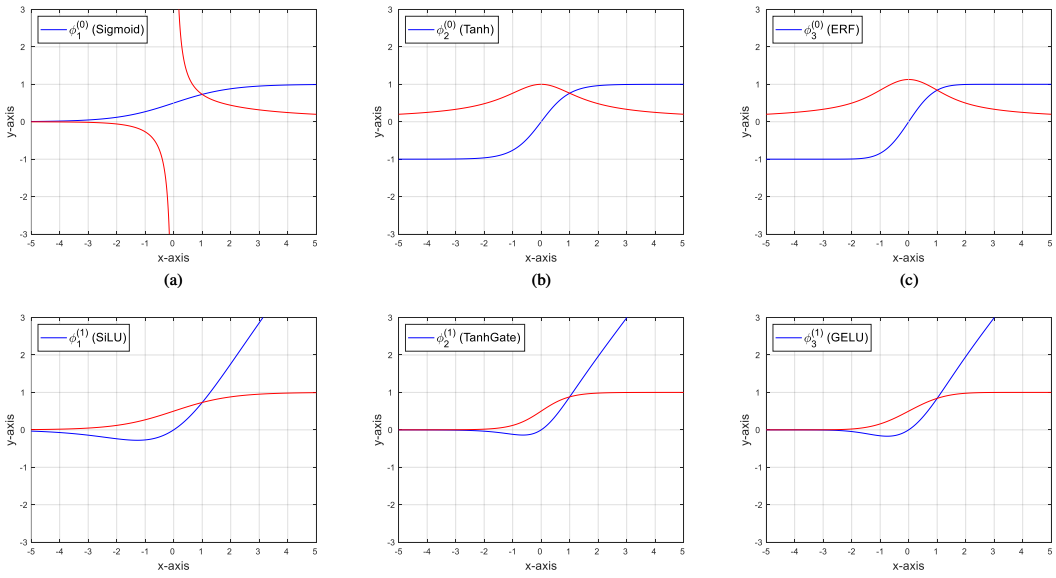


Figure 1: Illustrations of the (curves of) base functions $\left\{\phi_i^{(0)} \mid i = 1, 2, 3\right\}$ (top row) and their inspired self-gated functions $\left\{\phi_i^{(1)} \mid i = 1, 2, 3\right\}$ (bottom row). In each plot, the overall activation functions are colored by "**blue**" and the corresponding re-weighting functions are colored by "**red**," respectively. (a) $\phi_1^{(0)}$ (top) and $\phi_1^{(1)}$ (bottom); (b) $\phi_2^{(0)}$ (top) and $\phi_2^{(1)}$ (bottom); (c) $\phi_3^{(0)}$ (top) and $\phi_3^{(1)}$ (bottom).

where the superscript "(0)" represents "base form;" $\phi_1^{(0)}(\cdot)$ to $\phi_3^{(0)}(\cdot)$ denote $\mathrm{sigmoid}(\cdot)$, $\tanh(\cdot)$, and $\mathrm{erf}(\cdot)$, respectively. $\varsigma_1^{(0)}(\cdot)$ to $\varsigma_3^{(0)}(\cdot)$ are the corresponding re-weighting functions of the base functions. Note that $\varsigma_1^{(0)}(\tilde{x})$ (*i.e.*, the re-weighting function of Sigmoid function) has no finite definition at $\tilde{x} = 0$ (*i.e.*, its left-limit is $-\infty$ and right-limit is $+\infty$).

Note that for neural activation w.r.t. (visual) pattern recognition, these base functions $\left\{\phi_i^{(0)}(\tilde{x}) \mid i = 1, 2, 3\right\}$ are all *nonlinear* and *Lipschitz continuous*. Especially, Sigmoid function was the most prevailing activation function before ReLU [5]. That is, past interpretations/understandings cannot distinguish them by the expected activation abilities.

Figure 1(top row) depicts the curves of $\left\{\phi_i^{(0)}(\tilde{x})\right\}$ (colored by "**blue**") and $\left\{\varsigma_i^{(0)}(\tilde{x})\right\}$ (colored by "**red**"), respectively. They (their re-weighting functions) all violate **Property 4.1**. Therefore,

1. we expect that these base functions will all be inferior to ReLU in activation performance.

2. Particularly, $\phi_1^{(0)}(\tilde{x})$ (Sigmoid) violates Property 4.1 the comparatively most among them and it also violates Property 4.2 (CNI) and Property 4.3 (PPI), simultaneously. Therefore, we expect $\phi_1^{(0)}(\tilde{x})$ to show the comparatively most inferior performance among the base functions.

Table 1 reports the experimental results which validate the MCDM-hypothesis-based qualitative predictions 1 and 2.

Table 1: Experimental evaluation of the base functions and their self-gated functions. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1.

	Activation	Backbone	#Params.	Top-1(%)↑
Baseline	Linear (W/o Act)	CF-ResNet-56 [11]	0.6M	**16.5** ± 0.1
	ReLU [5]		0.6M	74.4 ± 0.3
Base	$\phi_1^{(0)}$ (Sigmoid)	CF-ResNet-56 [11]	0.6M	**46.5** ± 1.4
	$\phi_2^{(0)}$ (Tanh)		0.6M	72.3 ± 0.3
	$\phi_3^{(0)}$ (ERF)		0.6M	72.5 ± 0.2
Self-gated	$\phi_1^{(1)}$ (SiLU [13])	CF-ResNet-56 [11]	0.6M	75.3 ± 0.4
	$\phi_2^{(1)}$ (TanhGate Tab. 5.12)		0.6M	75.4 ± 0.3
	$\phi_3^{(1)}$ (GELU [12])		0.6M	75.3 ± 0.3
Reference	Mish [14]	CF-ResNet-56 [11]	0.6M	75.2 ± 0.3
	Pserf [8]		0.6M	75.3 ± 0.2
	SMU [8]		0.6M	74.9 ± 0.3
	SMU-1 [8]		0.6M	74.7 ± 0.2

Self-gated form.

$$\phi_i^{(1)}(\tilde{x}) = \varsigma_i^{(1)}(\tilde{x})\,\tilde{x}, i = 1, 2, 3\,, \tag{41}$$

which are the self-gated re-weighting functions built on the base functions. In particular, to meet Property 4.1, we let $\varsigma_1^{(1)}(\tilde{x}) = \mathrm{sigmoid}(\tilde{x})$, $\varsigma_2^{(1)}(\tilde{x}) = 0.5(\tanh(\tilde{x}) + 1)$, and $\varsigma_3^{(1)}(\tilde{x}) = 0.5(\mathrm{erf}(\tilde{x}/\sqrt{2}) + 1)$, respectively.

Figure 1(bottom row) depicts the curves of $\left\{\phi_i^{(1)}(\tilde{x})\right\}$ (colored by "**blue**") and $\left\{\varsigma_i^{(1)}(\tilde{x})\right\}$ (colored by "**red**"), respectively. Similar to ReLU, these self-gated functions all hold the 4 basic properties of MCDM hypothesis with the condition $\hat{\varrho}_x = \tilde{x}$. Therefore, we expect that they improve their base functions in activation ability.

As discussed in Section 5.1, soft-gated activation functions improve ReLU by introducing smoothing re-calibrations on the inputs. From the perspective of MCDM hypothesis, for different features, smooth soft-gated re-weighting functions provide more fine-grained importance scorings to distinguish the differences of influence of different $\hat{\varrho}_x$. Compared to the rigid binary masking (*i.e.*, the re-weighting function of ReLU), these smooth re-weighting functions are more consistent with the Intuition 4.2. Note that the rigid binary masking can be regarded as an extreme case of Property 4.1.

Then, from MCDM hypothesis, these 3 self-gated functions are similar in attributes to each other according to the basic intuitions and properties. Therefore, we expect that they will be close in activation performance.

Table 1 reports the experimental results, where

1. $\left\{\phi_i^{(1)}(\tilde{x})\right\}$ all outperform ReLU and the corresponding base functions $\left\{\phi_i^{(0)}(\tilde{x})\right\}$.
2. Moreover, these 3 self-gated functions are almost indistinguishable by activation performance.

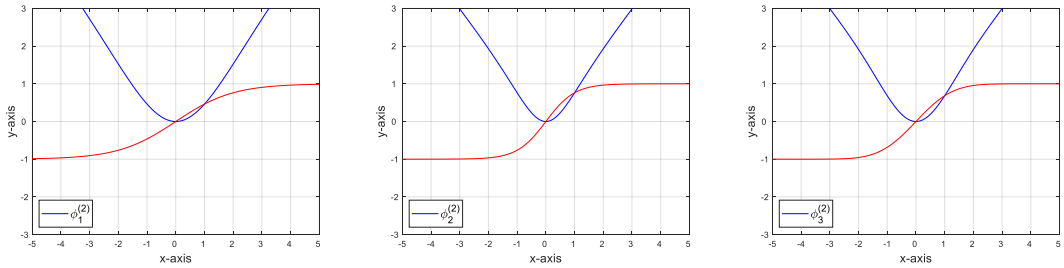


Figure 2: Illustrations of the (curves of) modified functions $\left\{\phi_i^{(2)} \mid i=1,2,3\right\}$ (colored by "**blue**") and their corresponding re-weighting functions $\left\{\varsigma_i^{(2)} \mid i=1,2,3\right\}$ (colored by "**red**). (a) $\phi_1^{(2)}$ and $\varsigma_1^{(2)}$; (b) $\phi_2^{(2)}$ and $\varsigma_2^{(2)}$; (c) $\phi_3^{(2)}$ and $\varsigma_3^{(2)}$.

3. Further, similarly, these 3 self-gated functions demonstrate close results to other popular/SOTA self-gated activation functions that also hold the 4 basic properties of MCDM hypothesis with the condition $\hat{\varrho}_x = \tilde{x}$.

These experimental phenomena are in line with our MCDM hypothesis.

In the subsequent, we discuss and validate the 4 basic properties inspired by MCDM hypothesis, respectively, by giving tailored modifications to the 3 self-gated functions to violate the corresponding basic properties and comparing the modified functions with their original self-gated functions.

Evaluation of Property 4.1 (The directional monotonicity $\wedge$ sign constraint of $\varsigma\left(\varrho_x\right)$ about ϱ_x).

$$\phi_i^{(2)}\left(\tilde{x}\right) = \varsigma_i^{(2)}\left(\tilde{x}\right)\tilde{x}, i=1,2,3\,. \tag{42}$$

We consider a simple case that violate Property 4.1, *i.e.*, $\varsigma_1^{(2)}\left(\tilde{x}\right) = 2\cdot\operatorname{sigmoid}\left(\tilde{x}\right)-1$, $\varsigma_2^{(2)}\left(\tilde{x}\right) = \tanh\left(\tilde{x}\right)$, and $\varsigma_3^{(2)}\left(\tilde{x}\right) = \operatorname{erf}\left(\tilde{x}/\sqrt{2}\right)$.

Figure 2 depicts the curves of $\left\{\phi_i^{(2)}\left(\tilde{x}\right)\right\}$ (colored by "**blue**") and $\left\{\varsigma_i^{(2)}\left(\tilde{x}\right)\right\}$ (colored by "**red**"), respectively.

Table 2: Evaluation on Property 4.1. We report the $\mathrm{mean}\pm\mathrm{std}$ of the Top-1.

	Activation	Backbone	#Params.	Top-1(%)↑
Baseline	Linear (W/o Act)	CF-ResNet-56 [11]	0.6M	**16.5** $\pm$ 0.1
	ReLU [5]		0.6M	74.4 $\pm$ 0.3
$\phi_1^{(1)}$-based	$\phi_1^{(1)}$ (SiLU [13])	CF-ResNet-56 [11]	0.6M	75.3 $\pm$ 0.4
	$\phi_1^{(2)}$		0.6M	**65.8** $\pm$ 1.9
$\phi_2^{(1)}$-based	$\phi_2^{(1)}$ (TanhGate Tab. 5.12)	CF-ResNet-56 [11]	0.6M	75.4 $\pm$ 0.3
	$\phi_2^{(2)}$		0.6M	**70.9** $\pm$ 0.3
$\phi_3^{(1)}$-based	$\phi_3^{(1)}$ (GELU [12])	CF-ResNet-56 [11]	0.6M	75.3 $\pm$ 0.3
	$\phi_3^{(2)}$		0.6M	**70.6** $\pm$ 1.2

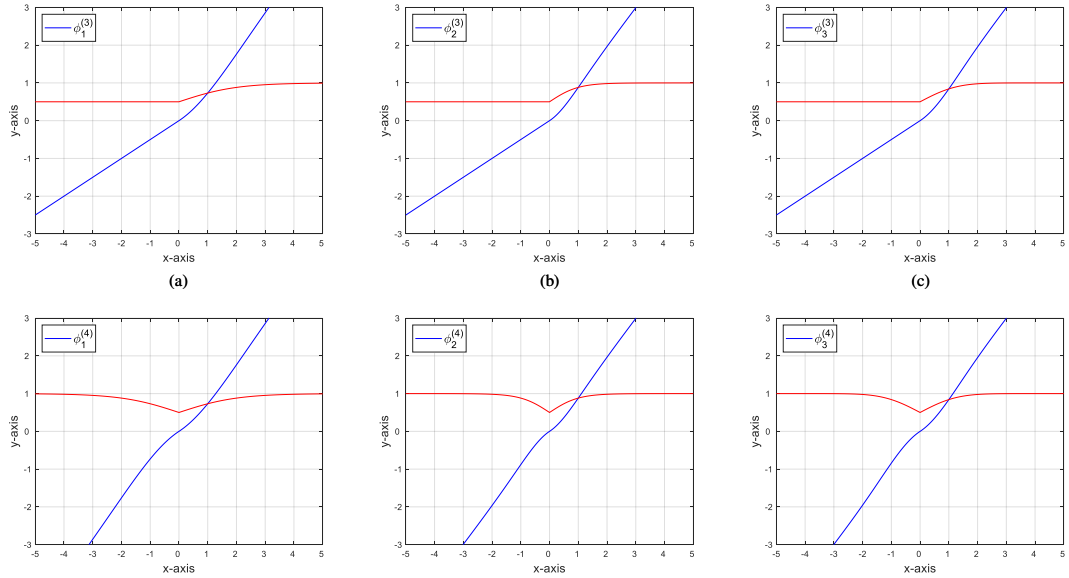


Figure 3: **Top row:** Illustrations of the (curves of) modified functions $\left\{\phi_i^{(3)} \mid i=1,2,3\right\}$ (colored by "**blue**") and their corresponding re-weighting functions $\left\{\varsigma_i^{(3)} \mid i=1,2,3\right\}$ (colored by "**red**). (a) $\phi_1^{(3)}$ and $\varsigma_1^{(3)}$; (b) $\phi_2^{(3)}$ and $\varsigma_2^{(3)}$; (c) $\phi_3^{(3)}$ and $\varsigma_3^{(3)}$. **Bottom row:** Illustrations of the (curves of) modified functions $\left\{\phi_i^{(4)} \mid i=1,2,3\right\}$ (colored by "**blue**") and their corresponding re-weighting functions $\left\{\varsigma_i^{(4)} \mid i=1,2,3\right\}$ (colored by "**red**). (a) $\phi_1^{(4)}$ and $\varsigma_1^{(4)}$; (b) $\phi_2^{(4)}$ and $\varsigma_2^{(4)}$; (c) $\phi_3^{(4)}$ and $\varsigma_3^{(4)}$.

Table 2 reports the experimental results, where all the three tailored kinds of modified functions that violate Property 4.1 demonstrate inferior results to their original self-gated re-weighting functions ($\left\{\phi_i^{(1)}\left(\tilde{x}\right)\right\}$). Note that as discussed above, the three base functions can also be included in this kind of activation functions and they demonstrate inferior activation performances to their self-gated functions.

These experimental results are consistent with our hypothesis, which validates the suggested Property 4.1.

Evaluation of Property 4.2 (CNI).

We consider 2 cases that violate Property 4.2 in different ways:

1. **Functions violating $\lim_{\hat{\varrho}_x \to -\infty} \left|\varsigma\left(\hat{\varrho}_x\right)\right| = 0$ by weight truncation.**

$$\phi_i^{(3)}\left(\tilde{x}\right) = \varsigma_i^{(3)}\left(\tilde{x}\right)\tilde{x}, i=1,2,3\,. \tag{43}$$

We consider a simple case where

$$\varsigma_i^{(3)}\left(\tilde{x}\right) = \begin{cases} \varsigma_i^{(1)}\left(0\right), & \tilde{x} < 0\,; \\ \varsigma_i^{(1)}\left(\tilde{x}\right), & \tilde{x} \geq 0\,. \end{cases} \tag{44}$$

Table 3: Experimental evaluation on Property 4.2. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1. "NaN" denotes failed training.

		Activation	Backbone	#Params.	Top-1(%)↑
Baseline		Linear (W/o Act)	CF-ResNet-56 [11]	0.6M	$\mathbf{16.5} \pm 0.1$
		ReLU [5]		0.6M	74.4 ± 0.3
$\phi_1^{(1)}$-based	Original	$\phi_1^{(1)}$ (SiLU)	CF-ResNet-56 [11]	0.6M	75.3 ± 0.4
	case1	$\phi_1^{(3)}$		0.6M	74.1 ± 0.2
	case2	$\phi_1^{(4)}$		0.6M	$\mathbf{43.2} \pm 27.4$
$\phi_2^{(1)}$-based	Original	$\phi_2^{(1)}$ (TanhGate)	CF-ResNet-56 [11]	0.6M	75.4 ± 0.3
	case1	$\phi_2^{(3)}$		0.6M	74.9 ± 0.3
	case2	$\phi_2^{(4)}$		0.6M	NaN
$\phi_3^{(1)}$-based	Original	$\phi_3^{(1)}$ (GELU)	CF-ResNet-56 [11]	0.6M	75.3 ± 0.3
	case1	$\phi_3^{(3)}$		0.6M	74.7 ± 0.2
	case2	$\phi_3^{(4)}$		0.6M	NaN

Figure 3(top row) depicts the curves of $\left\{\phi_i^{(3)}\left(\tilde{x}\right)\right\}$ (colored by "**blue**") and $\left\{\varsigma_i^{(3)}\left(\tilde{x}\right)\right\}$ (colored by "**red**"), respectively.

According to the physical meaning of Property 4.2, we expect that the performances of the modified functions will perform inferior to their original self-gated functions, respectively.

2. **Functions violating $\lim_{\hat{\varrho}_x \to -\infty} |\varsigma\left(\hat{\varrho}_x\right)| = 0$ by reversing the directional monotonicity on $\hat{\varrho}_x \in (-\infty, 0)$.**

$$\phi_i^{(4)}\left(\tilde{x}\right) = \varsigma_i^{(4)}\left(\tilde{x}\right)\tilde{x}, i = 1, 2, 3\,. \tag{45}$$

We consider the case

$$\varsigma_i^{(4)}\left(\tilde{x}\right) = \begin{cases} 2 \cdot \varsigma_i^{(1)}\left(-\tilde{x}\right)\,, & \tilde{x} < 0\,; \\ 2 \cdot \varsigma_i^{(1)}\left(\tilde{x}\right)\,, & \tilde{x} \geq 0\,. \end{cases} \tag{46}$$

Figure 3(bottom row) depicts the curves of $\left\{\phi_i^{(4)}\left(\tilde{x}\right)\right\}$ (colored by "**blue**") and $\left\{\varsigma_i^{(4)}\left(\tilde{x}\right)\right\}$ (colored by "**red**"), respectively.

According to the physical meaning of Property 4.2, similarly, we expect that the performances of the modified functions (1) will all perform inferior to their original self-gated functions, respectively; (2) each $\phi_i^{(4)}\left(\tilde{x}\right)$ will yield even inferior results to $\phi_i^{(3)}\left(\tilde{x}\right)$ that applies weight truncation as they violate Property 4.2 more seriously.

Results reported in Table 3 demonstrate the consistency of the experimental evaluations and the hypothetical qualitative assessments. This validates the suggested Property 4.2.

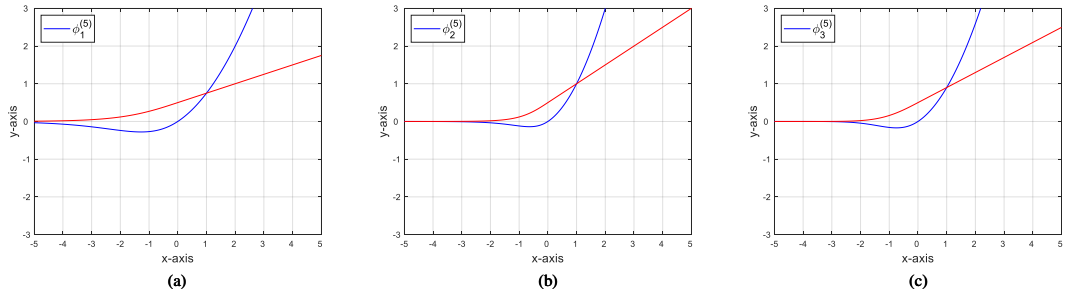


Figure 4: Illustrations of the (curves of) modified functions $\left\{\phi_i^{(5)} \mid i=1,2,3\right\}$ (colored by "**blue**") and their corresponding re-weighting functions $\left\{\varsigma_i^{(5)} \mid i=1,2,3\right\}$ (colored by "**red**). (a) $\phi_1^{(5)}$ and $\varsigma_1^{(5)}$; (b) $\phi_2^{(5)}$ and $\varsigma_2^{(5)}$; (c) $\phi_3^{(5)}$ and $\varsigma_3^{(5)}$.

Table 4: Experimental evaluation on Property 4.3. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1. "NaN" denotes failed training.

	Activation	Backbone	#Params.	Top-1(%)↑
Baseline	Linear (W/o Act)	CF-ResNet-56 [11]	0.6M	**16.5** $\pm$ 0.1
	ReLU [5]		0.6M	74.4 $\pm$ 0.3
$\phi_1^{(1)}$-based	$\phi_1^{(1)}$ (SiLU [13])	CF-ResNet-56 [11]	0.6M	75.3 $\pm$ 0.4
	$\phi_1^{(5)}$		0.6M	NaN
$\phi_2^{(1)}$-based	$\phi_2^{(1)}$ (TanhGate Tab. 5.12)	CF-ResNet-56 [11]	0.6M	75.4 $\pm$ 0.3
	$\phi_2^{(5)}$		0.6M	NaN
$\phi_3^{(1)}$-based	$\phi_3^{(1)}$ (GELU [12])	CF-ResNet-56 [11]	0.6M	75.3 $\pm$ 0.3
	$\phi_3^{(5)}$		0.6M	NaN

Evaluation of Property 4.3 (PPI).

$$\phi_i^{(5)}(\tilde{x}) = \varsigma_i^{(5)}(\tilde{x})\,\tilde{x}, i=1,2,3\,. \tag{47}$$

We consider a simple modification that makes each re-weighting function violate Property 4.3 by letting

$$\varsigma_i^{(5)}(\tilde{x}) = \begin{cases} \varsigma_i^{(1)}(\tilde{x})\,, & \tilde{x}<0\,; \\ \frac{\partial \varsigma_i^{(1)}(\tilde{x})}{\partial \tilde{x}}\mid_{\tilde{x}=0} \cdot \tilde{x} + \varsigma_i^{(1)}(0)\,, & \tilde{x}\geq 0\,. \end{cases} \tag{48}$$

Figure 4 depicts the curves of $\left\{\phi_i^{(5)}(\tilde{x})\right\}$ (colored by "**blue**") and $\left\{\varsigma_i^{(5)}(\tilde{x})\right\}$ (colored by "**red**"), respectively.

Table 4 shows the experimental results of the original self-gated functions ($\left\{\phi_i^{(1)}(\tilde{x})\right\}$) and their modified functions ($\left\{\phi_i^{(5)}(\tilde{x})\right\}$). Compared to $\phi_i^{(1)}(\tilde{x})$, each $\phi_i^{(5)}(\tilde{x})$ demonstrates a significant drop in accuracy, which is in line with the expectation. This validates the suggested Property 4.3.

Evaluation of Property 4.4 (OD).

$$\phi_i^{(6)}(\tilde{x}) = \varsigma_i^{(6)}(\tilde{x})\,\tilde{x}, i = 1, 2, 3\,. \tag{49}$$

We evaluate 3 simple cases, *i.e.*, $\varsigma_i^{(6)}(\tilde{x}) = \kappa_j \cdot \varsigma_i^{(1)}(\tilde{x}), j = 1, 2, 3$, where $\kappa_1 = 0.1$, $\kappa_2 = 0.5$, and $\kappa_3 = 1$ (*i.e.*, identity to the each original function).

Table 5: Evaluation on Property 4.4. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1.

	Activation	Backbone	#Params.	Top-1(%)↑
Baseline	Linear (W/o Act)	CF-ResNet-56 [11]	0.6M	**16.5** $\pm$ 0.1
	ReLU [5]		0.6M	74.4 $\pm$ 0.3
$\kappa_1 = 0.1$	$\phi_1^{(6)}$	CF-ResNet-56 [11]	0.6M	66.5 $\pm$ 0.5
$\kappa_2 = 0.5$			0.6M	72.2 $\pm$ 0.3
$\kappa_3 = 1$ ($\phi_1^{(1)}$)			0.6M	75.3 $\pm$ 0.4
$\kappa_1 = 0.1$	$\phi_2^{(6)}$	CF-ResNet-56 [11]	0.6M	68.5 $\pm$ 0.2
$\kappa_2 = 0.5$			0.6M	73.8 $\pm$ 0.6
$\kappa_3 = 1$ ($\phi_2^{(1)}$)			0.6M	75.4 $\pm$ 0.3
$\kappa_1 = 0.1$	$\phi_3^{(6)}$	CF-ResNet-56 [11]	0.6M	67.9 $\pm$ 0.4
$\kappa_2 = 0.5$			0.6M	73.7 $\pm$ 0.4
$\kappa_3 = 1$ ($\phi_3^{(1)}$)			0.6M	75.3 $\pm$ 0.3

Based on the physical meaning of Property 4.4, we expect that

1. the modification of κ_1 will lead to significant decreases in accuracy to all the modified functions;
2. the modified functions corresponding to κ_2 will have comparatively closer activation performances to the original self-gated functions than the counterparts with κ_1.

Note that as discussed (Section .4), Property 4.4 is a relaxed constraint since the trainable parameters of a (recent) neural network layer (*e.g.*, linear layer and parametric normalization layer) are capable of providing an extent of flexibility to the (intensities of) features.

The comparative results reported in Table 5 demonstrate the consistency of the experimental evaluations and the hypothetical qualitative assessments.

The experimental evaluation results are consistent with the hypothesis. This validates the suggested Property 4.4.

Evaluation of Intuition 4.3 (Mismatched Feature Scoring)

As discussed in Section 5.1, popular/SOTA self-gated activation functions demonstrate clear improvements to ReLU by leveraging smooth re-weighting. Their capability of activation, however, can still be limited by the critical *mismatched feature scoring* problem which we infer from MCDM hypothesis and otherwise invisible to past explanations.

Table 6: Evaluation on Intuition 4.3. We report the $\mathrm{mean} \pm \mathrm{std}$ of the Top-1.

	Activation	Backbone	#Params.	Top-1(%)↑
Baseline	Linear (W/o Act)	CF-ResNet-56 [11]	0.6M	**16.5** $\pm$ 0.1
	ReLU [5]		0.6M	74.4 $\pm$ 0.3
$\phi_1^{(1)}$-based	$\phi_1^{(1)}$ (SiLU [13])	CF-ResNet-56 [11]	0.6M	75.3 $\pm$ 0.4
	$\phi_1^{(7)}$ (SiLU-Ada)		0.6M	**76.5** $\pm$ 0.3
$\phi_2^{(1)}$-based	$\phi_2^{(1)}$ (TanhGate Tab. 5.12)	CF-ResNet-56 [11]	0.6M	75.4 $\pm$ 0.3
	$\phi_2^{(7)}$ (TanhGate-Ada)		0.6M	**76.5** $\pm$ 0.3
$\phi_3^{(1)}$-based	$\phi_3^{(1)}$ (GELU [12])	CF-ResNet-56 [11]	0.6M	75.3 $\pm$ 0.3
	$\phi_3^{(7)}$ (GELU-Ada)		0.6M	**76.3** $\pm$ 0.2

Therefore, we expect that the original self-gated functions $\left\{\phi_i^{(1)}\left(\tilde{x}\right)\right\}$ will be improved by giving AdaShift-style modifications, *i.e.*, by replacing the common approximated similarity $\hat{\varrho}\left(\tilde{x}\right) = \tilde{x}$ with the AdaShift-style approximated similarity $\hat{\varrho}\left(\tilde{x}\right) = \tilde{x} + \Delta$, *i.e.*,

$$\phi_i^{(7)}\left(\tilde{x}\right) = \varsigma_i^{(6)}\left(\tilde{x} + \Delta\right)\tilde{x}, i = 1, 2, 3\,. \tag{50}$$

For simplicity, we consider incorporating the basic Δ of AdaShift-B (defined by Equation (5.8)) to $\left\{\phi_i^{(1)}\left(\tilde{x}\right)\right\}$.

The experimental results are reported in Table 6, where we demonstrate that our vanilla and the corresponding modified AdaShift-B(s) improve different self-gated activation function counterparts significantly and consistently. Note that in the preceding ablation study (Section 5.5.3), *i.e.*, "Generalizing AdaShift by varying re-weighting function," we also validate the generalizability of our AdaShift prototype to Mish's [14] re-weighting functions (demonstrated in Table 5.12), a relevant function to the adopted base functions. This validates our Intuition 4.3 for improving neural activation.

It is worth noting that most of our experimental results demonstrated in Sections 4.5 and 5.5 are evidence for Intuition 4.3 because this intuition serves as one of the fundamental clues for us to propose our novel activation prototypes IIEU and AdaShift.

References

[1] S. Cai, R. Wakaki, S. Nobuhara, and K. Nishino, "Rgb road scene material segmentation," in *Proc. Asian Conference on Computer Vision (ACCV)*, 2022. [Cited on pages iii, ix, x, xi, xv, xvii, xviii, 1, 4, 20, 25, 26, 27, 29, 38, 39, 41, 68, 78, 89, 101, and 102]

[2] S. Cai, "Iieu: Rethinking neural feature activation from decision-making," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 5796–5806, 2023. [Cited on pages iii, xi, xii, xiii, xvi, xvii, 1, 54, 57, 64, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 80, 82, 89, 91, 92, 93, 96, and 101]

[3] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 833–851, 2018. [Cited on pages xi, 20, 22, 23, 24, 36, 37, 38, 39, 42, 46, 47, 49, and 51]

[4] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and Efficient Design for Semantic Segmentation with Transformers," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [Cited on pages xi, xv, xvi, 20, 21, 22, 24, 28, 36, 37, 38, 39, 42, 43, 45, 46, 47, 49, 51, 78, 101, and 102]

[5] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. International Conference on Machine Learning (ICML)*, 2010. [Cited on pages xi, xii, 3, 5, 53, 54, 63, 66, 67, 68, 70, 71, 72, 75, 77, 78, 79, 89, 91, 92, 93, 94, 96, 97, 99, 100, 101, 102, 124, 125, 126, 128, 129, 130, and 131]

[6] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," in *Proc. Workshop Track of the 6th International Conference on Learning Representations (ICLR)*, 2018. [Cited on pages xi, 54, 67, 68, 70, 71, 75, 77, 78, 82, 89, 91, 92, 94, 96, 97, 98, 101, and 102]

[7] N. Ma, X. Zhang, M. Liu, and J. Sun, "Activate or not: Learning customized activation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8032–8042, 2021. [Cited on pages xi, xiii, xvi, 5, 53, 54, 67, 68, 69, 70, 71, 72, 75, 77, 78, 80, 82, 86, 89, 91, 92, 93, 96, 97, 101, and 102]

[8] K. Biswas, S. Kumar, S. Banerjee, and A. K. Pandey, "Smooth maximum unit: Smooth activation function for deep networks using smoothing maximum technique," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*

(CVPR), 2022. [Cited on pages xi, xiii, xvi, 5, 53, 54, 67, 68, 69, 70, 71, 72, 75, 77, 78, 80, 82, 89, 91, 92, 93, 94, 96, 97, 101, 102, and 125]

[9] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, 2018. [Cited on pages xi, xvi, xvii, 54, 68, 70, 71, and 72]

[10] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 116–131, 2018. [Cited on pages xi, xvi, 54, 68, 70, and 71]

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. [Cited on pages xi, xiii, xvii, xviii, 36, 37, 39, 47, 49, 54, 68, 71, 72, 75, 77, 78, 89, 90, 91, 92, 93, 94, 96, 97, 99, 100, 101, 102, 122, 123, 125, 126, 128, 129, 130, and 131]

[12] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *arXiv preprint arXiv:1606.08415*, 2016. [Cited on pages xii, 5, 11, 30, 53, 67, 68, 70, 75, 80, 82, 84, 89, 91, 92, 95, 96, 99, 100, 125, 126, 129, and 131]

[13] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Networks*, vol. 107, pp. 3–11, 2018. [Cited on pages xii, 67, 68, 75, 80, 82, 84, 86, 89, 91, 92, 96, 97, 99, 100, 125, 126, 129, and 131]

[14] D. Misra, "Mish: A self regularized non-monotonic neural activation function," in *Proc. British Machine Vision Conference (BMVC)*, 2020. [Cited on pages xii, 11, 67, 68, 70, 71, 75, 80, 82, 84, 89, 91, 92, 96, 100, 125, and 131]

[15] K. Biswas, S. Kumar, S. Banerjee, and A. K. Pandey, "Erfact and pserf: Non-monotonic smooth trainable activation functions," in *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, 2022. [Cited on pages xiii, xvi, 5, 53, 67, 68, 69, 70, 72, 80, 82, 89, 91, 92, 94, and 98]

[16] Y. Zhou, Z. Zhu, and Z. Zhong, "Learning specialized activation functions with the piecewise linear unit," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 12095–12104, 2021. [Cited on pages xiii, 67, 68, 69, 70, 72, and 93]

[17] J. Zhou, V. Jampani, Z. Pi, Q. Liu, and M.-H. Yang, "Decoupled Dynamic Filter Networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [Cited on pages xiii, xvii, 36, 37, 39, 69, 70, 71, 72, 74, and 89]

[18] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 42, no. 8, pp. 2011–2023, 2020. [Cited on pages xiii, 1, 24, 66, 68, 69, 70, 71, 82, 93, and 97]

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *Proc. International Conference on Learning Representations (ICLR)*, 2021. [Cited on pages xv, xvi, xviii, 1, 10, 23, 36, 37, 39, 49, 89, 94, and 95]

[20] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "CvT: Introducing Convolutions to Vision Transformers," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 22–31, 2021. [Cited on pages xv, xvi, 24, 36, 37, 39, and 49]

[21] M. Ding, B. Xiao, N. Codella, P. Luo, J. Wang, and L. Yuan, "Davit: Dual attention vision transformers," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 74–92, Springer, 2022. [Cited on pages xv, xvi, 21, 24, 28, 36, 37, 39, and 49]

[22] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11976–11986, 2022. [Cited on pages xv, xvi, xviii, 21, 24, 28, 36, 37, 39, 49, 89, 94, and 95]

[23] T.-Y. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 740–755, 2014. [Cited on pages xvii, xviii, 22, 68, 77, 89, and 101]

[24] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018. [Cited on pages xvii and 77]

[25] W. Yu, M. Luo, P. Zhou, C. Si, Y. Zhou, X. Wang, J. Feng, and S. Yan, "Metaformer is actually what you need for vision," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [Cited on pages xviii, 1, 24, 94, and 95]

[26] 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 *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2021. [Cited on pages xviii, 1, 23, 94, and 95]

[27] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, *et al.*, "Mlp-mixer: An all-mlp architecture for vision," *Advances in neural information processing systems*, vol. 34, pp. 24261–24272, 2021. [Cited on pages xviii and 95]

[28] A. Krizhevsky, "Learning multiple layers of features from tiny images," Master's thesis, University of Toronto, 2009. [Cited on pages xviii, 68, 77, 89, 96, 97, 99, and 123]

[29] Weiaicunzai, "pytorch-cifar100." `https://github.com/weiaicunzai/pytorch-cifar100`. [Cited on pages xviii, 73, 77, 95, 96, 97, 100, and 123]

[30] B. Cheng, A. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [Cited on pages 1, 5, 53, and 79]

[31] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-YOLOv4: Scaling cross stage partial network," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [Cited on pages 1, 5, 53, and 79]

[32] G. Schwartz and K. Nishino, "Recognizing Material Properties from Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 42, no. 8, pp. 1981–1995, 2020. [Cited on pages 1, 5, 22, 53, and 79]

[33] S. Cai, Y. Guo, S. Khan, J. Hu, and G. Wen, "Ground-to-aerial image geo-localization with a hard exemplar reweighting triplet loss," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2019. [Cited on pages 1, 5, 53, and 79]

[34] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 551–561, 2016. [Cited on pages 1 and 10]

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All You Need," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, pp. 5998–6008, 2017. [Cited on pages 1, 10, and 23]

[36] Y. Han, G. Huang, S. Song, L. Yang, H. Wang, and Y. Wang, "Dynamic neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 7436–7456, 2021. [Cited on page 1]

[37] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," in *Proc. International Conference on Learning Representations (ICLR)*, pp. 1–11, 2021. [Cited on pages 1, 24, and 30]

[38] C. Dugas, Y. Bengio, F. Bélisle, C. Nadeau, and R. Garcia, "Incorporating second-order functional knowledge for better option pricing," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2000. [Cited on pages 3, 5, 53, 66, 68, 79, 89, 91, and 92]

[39] T. Serre, A. Oliva, and T. Poggio, "A feedforward architecture accounts for rapid categorization," *PNAS*, vol. 104, no. 15, pp. 6424–6429, 2007. [Cited on pages 3, 5, 53, and 79]

[40] M. Kouh, *Toward a more biologically plausible model of object recognition*. PhD thesis, MIT, 2007. [Cited on pages 3, 5, 53, and 79]

[41] N. Ma, X. Zhang, and J. Sun, "Funnel activation for visual recognition," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 351–368, Springer, 2020. [Cited on pages 5, 53, 68, 70, 71, 89, and 93]

[42] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu, "Dynamic relu," in *Proc. European Conference on Computer Vision (ECCV)*, 2020. [Cited on pages 5, 53, 68, 89, and 93]

[43] L. Meronen, M. Trapp, and A. Solin, "Periodic Activation Functions Induce Stationarity," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [Cited on pages 5 and 53]

[44] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein, "Implicit Neural Representations with Periodic Activation Functions," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2020. [Cited on pages 5 and 53]

[45] A. Molina, P. Schramowski, and K. Kersting, "Padé Activation Units: End-to-end Learning of Flexible Activation Functions in Deep Networks," in *Proc. International Conference on Learning Representations (ICLR)*, 2020. [Cited on pages 5, 53, 67, 68, 79, and 89]

[46] P. Awasthi, A. Tang, and A. Vijayaraghavan, "Efficient algorithms for learning depth-2 neural networks with general relu activations," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [Cited on pages 5, 53, and 79]

[47] M. Goyal, R. Goyal, and B. Lall, "Learning activation functions: A new paradigm for understanding neural networks," *arXiv preprint arXiv:1906.09529*, 2019. [Cited on pages 5 and 53]

[48] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998. [Cited on page 9]

[49] J. K. Tsotsos, S. M. Culhane, W. Y. K. Wai, Y. Lai, N. Davis, and F. Nuflo, "Modeling visual attention via selective tuning," *Artificial intelligence*, vol. 78, no. 1-2, pp. 507–545, 1995. [Cited on page 9]

[50] M. Hayhoe and D. Ballard, "Eye movements in natural behavior," *Trends in cognitive sciences*, vol. 9, no. 4, pp. 188–194, 2005. [Cited on page 9]

[51] M. Corbetta and G. L. Shulman, "Control of goal-directed and stimulus-driven attention in the brain," *Nature Reviews Neuroscience*, vol. 3, no. 3, pp. 201–215, 2002. [Cited on page 9]

[52] E. Niebur, "Computational architectures for attention," *The attentive brain*, 1998. [Cited on page 9]

[53] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Computational visual media*, vol. 8, no. 3, pp. 331–368, 2022. [Cited on page 9]

[54] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-Local Neural Networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7794–7803, 2018. [Cited on pages 10 and 23]

[55] M. El Alaoui, *Fuzzy TOPSIS: Logic, Approaches, and Case Studies*. CRC Press, 2021. [Cited on page 12]

[56] C.-L. Hwang and K. Yoon, "Multiple attribute decision making methods and applications a state-of-the-art survey," [Cited on page 12]

[57] K. Yoon, "A reconciliation among discrete compromise solutions," *Journal of the Operational Research Society*, vol. 38, pp. 277–286, 1987. [Cited on page 12]

[58] C.-L. Hwang, Y.-J. Lai, and T.-Y. Liu, "A new approach for multiple objective decision making," *Computers & operations research*, vol. 20, no. 8, pp. 889–899, 1993. [Cited on page 12]

[59] D. Walczak and A. Rutkowska, "Project rankings for participatory budget based on the fuzzy topsis method," *European Journal of Operational Research*, vol. 260, no. 2, pp. 706–714, 2017. [Cited on page 12]

[60] X. Mi, M. Tang, H. Liao, W. Shen, and B. Lev, "The state-of-the-art survey on integrations and applications of the best worst method in decision making: Why, what, what for and what's next?," *Omega*, vol. 87, pp. 205–225, 2019. [Cited on page 12]

[61] P. Chen, "Effects of the entropy weight on topsis," *Expert Systems with Applications*, vol. 168, p. 114186, 2021. [Cited on pages 12 and 14]

[62] Y. Wang, P. Liu, and Y. Yao, "Bmw-topsis: A generalized topsis model based on three-way decision," *Information sciences*, vol. 607, pp. 799–818, 2022. [Cited on page 12]

[63] S. Corrente and M. Tasiou, "A robust topsis method for decision making problems with hierarchical and non-monotonic criteria," *Expert Systems with Applications*, vol. 214, p. 119045, 2023. [Cited on page 12]

[64] H. Li, J. Huang, Y. Hu, S. Wang, J. Liu, and L. Yang, "A new tmy generation method based on the entropy-based topsis theory for different climatic zones in china," *Energy*, vol. 231, p. 120723, 2021. [Cited on page 12]

[65] M. Lin, Z. Chen, Z. Xu, X. Gou, and F. Herrera, "Score function based on concentration degree for probabilistic linguistic term sets: an application to topsis and vikor," *Information Sciences*, vol. 551, pp. 270–290, 2021. [Cited on page 12]

[66] S.-M. Chen, S.-H. Cheng, and T.-C. Lan, "Multicriteria decision making based on the topsis method and similarity measures between intuitionistic fuzzy values," *Information Sciences*, vol. 367, pp. 279–295, 2016. [Cited on pages 12 and 53]

[67] D. Joshi and S. Kumar, "Interval-valued intuitionistic hesitant fuzzy choquet integral based topsis method for multi-criteria group decision making," *European Journal of Operational Research*, vol. 248, no. 1, pp. 183–191, 2016. [Cited on pages 12, 53, 56, and 83]

[68] X. Wu and F. Hu, "Analysis of ecological carrying capacity using a fuzzy comprehensive evaluation method," *Ecological Indicators*, vol. 113, p. 106243, 2020. [Cited on page 14]

[69] H. Li, G. Liu, and Z. Yang, "Improved gray water footprint calculation method based on a mass-balance model and on fuzzy synthetic evaluation," *Journal of Cleaner Production*, vol. 219, pp. 377–390, 2019. [Cited on page 14]

[70] S. Feng and L. D. Xu, "Decision support for fuzzy comprehensive evaluation of urban development," *Fuzzy Sets and Systems*, vol. 105, no. 1, pp. 1–12, 1999. [Cited on page 14]

[71] Y.-W. Du, S.-S. Wang, and Y.-M. Wang, "Group fuzzy comprehensive evaluation method under ignorance," *Expert systems with applications*, vol. 126, pp. 92–111, 2019. [Cited on page 15]

[72] L. A. Zadeh, "Fuzzy sets," *Information and control*, vol. 8, no. 3, pp. 338–353, 1965. [Cited on page 16]

[73] J.-L. Deng, "Control problems of grey systems," *Systems & control letters*, vol. 1, no. 5, pp. 288–294, 1982. [Cited on page 16]

[74] P. Liu, Y. Fu, P. Wang, and X. Wu, "Grey relational analysis-and clustering-based opinion dynamics model in social network group decision making," *Information Sciences*, vol. 647, p. 119545, 2023. [Cited on page 16]

[75] T. Škrinjarić, "Dynamic portfolio optimization based on grey relational analysis approach," *Expert systems with applications*, vol. 147, p. 113207, 2020. [Cited on page 16]

[76] L. Weng, Q. Zhang, Z. Lin, and L. Wu, "Harnessing heterogeneous social networks for better recommendations: A grey relational analysis approach," *Expert Systems with Applications*, vol. 174, p. 114771, 2021. [Cited on page 16]

[77] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, "Unified perceptual parsing for scene understanding," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 418–434, 2018. [Cited on pages 20, 24, 46, and 47]

[78] 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 *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6881–6890, 2021. [Cited on pages 20, 46, and 47]

[79] S. Bell, P. Upchurch, N. Snavely, and K. Bala, "Material Recognition in the Wild with the Materials in Context Database," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3479–3487, 2015. [Cited on pages 22, 36, 37, and 39]

[80] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1097–1105, 2012. [Cited on page 22]

[81] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015. [Cited on pages 22 and 39]

[82] P. Kraehenbuehl and V. Koltun, "Parameter Learning and Convergent Inference for Dense Random Fields," in *Proc. International Conference on Machine Learning (ICML)*, pp. 513–521, 2013. [Cited on page 22]

[83] G. Schwartz and K. Nishino, "Visual Material Traits: Recognizing Per-Pixel Material Context," in *IEEE Color and Photometry in Computer Vision Workshop*, 2013. [Cited on page 22]

[84] G. Schwartz and K. Nishino, "Automatically Discovering Local Visual Material Attributes," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. [Cited on page 22]

[85] G. Schwartz and K. Nishino, "Material Recognition from Local Appearance in Global Context.," *arXiv preprint arXiv:1611.09394*, 2016. [Cited on pages 22, 36, 37, and 39]

[86] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. [Cited on pages 22, 68, and 89]

[87] H. Zhang, K. Dana, and K. Nishino, "Reflectance Hashing for Material Recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3071–3080, 2015. [Cited on page 22]

[88] J. Xue, H. Zhang, K. Dana, and K. Nishino, "Differential Angular Imaging for Material Recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6940–6949, 2017. [Cited on page 22]

[89] H. Zhang, J. Xue, and K. Dana, "Deep Ten: Texture Encoding Network," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2896–2905, 2017. [Cited on page 22]

[90] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher Kernel for Large-Scale Image Classification," in *Proc. European Conference on Computer Vision (ECCV)*, vol. 6314, pp. 143–156, 2010. [Cited on page 22]

[91] J. Xue, H. Zhang, K. Nishino, and K. Dana, "Differential Viewpoints for Ground Terrain Material Recognition.," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pp. 1–1, 2020. [Cited on page 22]

[92] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, and R. Raska, "Deepglobe 2018: A Challenge to Parse the Earth Through Satellite Images," in *Proc. CVPR Workshop*, pp. 172–17209, 2018. [Cited on page 22]

[93] M. Purri, J. Xue, K. J. Dana, M. J. Leotta, D. Lipsa, Z. Li, B. Xu, and J. Shan, "Material Segmentation of Multi-View Satellite Imagery.," *arXiv preprint arXiv:1904.08537*, 2019. [Cited on page 22]

[94] M. Brown, H. Goldberg, K. Foster, A. Leichtman, S. Wang, S. Hagstrom, M. Bosch, and S. Almes, "Large-Scale Public Lidar and Satellite Image Data Set for Urban Semantic Labeling," in *Laser Radar Technology and Applications XXIII*, vol. 10636, 2018. [Cited on page 22]

[95] J. Xue, M. Purri, and K. Dana, "Angular luminance for material segmentation," in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020. [Cited on page 22]

[96] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. [Cited on pages 22, 24, 36, 37, 39, 47, and 49]

[97] G. Neuhold, T. Ollmann, S. R. Bulo, and P. Kontschieder, "The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 5000–5009, 2017. [Cited on page 22]

[98] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," *arXiv preprint arXiv:1706.05587*, 2017. [Cited on pages 22, 24, 36, 37, 39, 47, and 49]

[99] T.-J. Yang, M. D. Collins, Y. Zhu, J.-J. Hwang, T. Liu, X. Zhang, V. Sze, G. Papandreou, and L.-C. Chen, "Deeperlab: Single-Shot Image Parser," *arXiv preprint arXiv:1902.05093*, 2019. [Cited on pages 22, 24, 36, 37, and 39]

[100] S. Choi, J. T. Kim, and J. Choo, "Cars Can't Fly Up in the Sky: Improving Urban-Scene Segmentation via Height-Driven Attention networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9373–9383, 2020. [Cited on page 22]

[101] Y. Zhu, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam, A. Tao, and B. Catanzaro, "Improving Semantic Segmentation via Video Propagation and Label Relaxation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8856–8865, 2019. [Cited on page 22]

[102] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro, "SDC-Net: Video Prediction Using Spatially-Displaced Convolution," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 747–763, 2018. [Cited on page 22]

[103] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, "Stand-Alone Self-Attention in Vision Models," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, pp. 68–80, 2019. [Cited on pages 22 and 23]

[104] H. Wang, Y. Zhu, B. Green, H. Adam, A. L. Yuille, and L.-C. Chen, "Axial-Deeplab: Stand-Alone Axial-Attention for Panoptic Segmentation," in *Proc. European Conference on Computer Vision (ECCV)*, pp. 108–126, 2020. [Cited on pages 22 and 23]

[105] 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 *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3213–3223, 2016. [Cited on pages 22 and 51]

[106] Y. Liang, R. Wakaki, S. Nobuhara, and K. Nishino, "Multimodal Material Segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. [Cited on pages 23, 37, and 47]

[107] H. Zhang, I. J. Goodfellow, D. N. Metaxas, and A. Odena, "Self-Attention Generative Adversarial Networks," in *Proc. International Conference on Machine Learning (ICML)*, pp. 7354–7363, 2018. [Cited on page 23]

[108] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-Cross Attention for Semantic Segmentation," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 603–612, 2019. [Cited on page 23]

[109] H. Zhang, H. Zhang, C. Wang, and J. Xie, "Co-Occurrent Features in Semantic Segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 548–557, 2019. [Cited on page 23]

[110] 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 *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2021. [Cited on page 23]

[111] B. Graham, A. El-Nouby, H. Touvron, P. Stock, A. Joulin, H. Jégou, and M. Douze, "Levit: A Vision Transformer in Convnet's Clothing for Faster Inference," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2021. [Cited on page 23]

[112] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in Transformer," *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [Cited on page 23]

[113] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to Scale: Scale-Aware Semantic Image Segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3640–3649, 2016. [Cited on page 24]

[114] X. Li, W. Wang, X. Hu, and J. Yang, "Selective Kernel Networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 510–519, 2019. [Cited on pages 24, 36, 37, 39, 66, 70, 71, 74, and 95]

[115] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 40, no. 4, pp. 834–848, 2018. [Cited on page 24]

[116] D. Han, J. Kim, and J. Kim, "Deep pyramidal residual networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6307–6315, 2017. [Cited on page 24]

[117] J. Gu, H. Kwon, D. Wang, W. Ye, M. Li, Y.-H. Chen, L. Lai, V. Chandra, and D. Z. Pan, "Multi-scale high-resolution vision transformer for semantic segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12094–12103, 2022. [Cited on pages 24 and 30]

[118] A. Geiger, P. Lenz, and R. Urtasun, "Are We Ready for Autonomous Driving? The Kitti Vision Benchmark Suite," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. [Cited on page 26]

[119] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A Large-Scale Hierarchical Image Database," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009. [Cited on page 37]

[120] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal, "Context Encoding for Semantic Segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7151–7160, 2018. [Cited on page 37]

[121] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10781–10790, 2020. [Cited on page 43]

[122] Y. Xu, Y. Li, L. Zheng, L. Cui, S. Li, W. Li, and Y. Cai, "Site selection of wind farms using gis and multi-criteria decision making method in wafangdian, china," *Energy*, vol. 207, p. 118222, 2020. [Cited on pages 53, 56, and 83]

[123] J. Rezaei, "Best-worst multi-criteria decision-making method: Some properties and a linear model," *Omega-International Journal of Management Science*, vol. 64, pp. 126–130, 2016. [Cited on pages 53, 56, and 83]

[124] J. Qin, X. Liu, and W. Pedrycz, "An extended todim multi-criteria group decision making method for green supplier selection in interval type-2 fuzzy environment," *European Journal of Operational Research*, vol. 258, no. 2, pp. 626–638, 2017. [Cited on pages 53, 56, and 83]

[125] Y. Dong, Y. Liu, H. Liang, F. Chiclana, and E. Herrera-Viedma, "Strategic weight manipulation in multiple attribute decision making," *Omega-International Journal of Management Science*, vol. 75, pp. 154–164, 2018. [Cited on page 53]

[126] H.-B. Yan, T. Ma, and V.-N. Huynh, "On qualitative multi-attribute group decision making and its consensus measure: A probability based perspective," *Omega-International Journal of Management Science*, vol. 70, pp. 94–117, 2017. [Cited on page 53]

[127] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proc. International Conference on Machine Learning (ICML)*, pp. 448–456, 2015. [Cited on pages 55, 58, 62, 77, 83, and 114]

[128] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016. [Cited on pages 55, 58, 62, 66, 76, 77, 81, and 83]

[129] A. L. Maas, A. Y. Hannun, A. Y. Ng, *et al.*, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML Workshop*, 2013. [Cited on pages 66, 68, 82, 89, 91, and 92]

[130] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, 2015. [Cited on pages 67, 68, 71, 75, 82, 89, 93, and 96]

[131] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," in *Proc. International Conference on Learning Representations (ICLR)*, 2016. [Cited on pages 67, 68, 75, 82, 89, 91, 92, and 96]

[132] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *Proc. International Conference on Machine Learning (ICML)*, 2013. [Cited on pages 67 and 79]

[133] C. Brezinski and J. Van Iseghem, "Padé approximations," *Handbook of Numerical Analysis*, vol. 3, pp. 47–222, 1994. [Cited on page 67]

[134] L. Wu, "Learning a Single Neuron for Non-monotonic Activation Functions," in *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, PMLR, 2022. [Cited on pages 68 and 80]

[135] S. Cho, W. Chang, G. Lee, and J. Choi, "Interpreting internal activation patterns in deep temporal neural networks by finding prototypes," in *Proc. ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2021. [Cited on page 68]

[136] F. Agostinelli, M. Hoffman, P. Sadowski, and P. Baldi, "Learning activation functions to improve deep neural networks," in *Proc. International Conference on Learning Representations (ICLR)*, 2015. [Cited on pages 68 and 89]

[137] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications.," *arXiv preprint arXiv:1704.04861*, 2017. [Cited on page 70]

[138] Y. Wu and K. He, "Group normalization," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018. [Cited on page 77]

[139] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "MMDetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019. [Cited on pages 77 and 101]