SPECTRAL DOMAIN CONVOLUTIONAL NEURAL NETWORK OPTIMIZED FOR COMPUTATIONAL WORKLOAD AND MEMORY ACCESS COST

SHAHRIYAR MASUD RIZVI

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

MARCH 2023

DEDICATION

To my wife, Nusrat Mohaimen, without whose support I could not have pursued this academic journey.

ACKNOWLEDGEMENT

I am grateful to the almighty Allah SWT for bestowing me with the skills and tenacity to pursue this difficult yet rewarding journey in the face of many challenges and hardships. All praise is due to Allah, The Most Gracious and The Most Merciful. I would like to express my appreciation to the people who have supported me during this journey or contributed to my work in a direct or indirect manner.

I would like to express my sincere appreciation to my thesis supervisor Dr. Ab Al-Hadi Ab Rahman, who provided me with constant support and guidance throughout my Ph.D. journey. I am grateful for his patience and his insights regarding my research work and directions. I am sincerely thankful to my co-supervisor Dr. Usman Ullah Sheikh for providing his valuable comments when reviewing this thesis. I am grateful for his encouraging words. I would like to acknowledge my former co-supervisor Prof. Dr. Mohamed Khalil-Hani, who at times treated me as a son, for his support and encouragement, during the early years of my Ph.D. journey.

I would like to thank my colleagues at the VeCAD laboratory, Dr. Sayed Omid Ayat and Dr. Hamdan Abdellatef, for their valuable suggestions and support. I would like to thank Dr. Mohammed Sultan Ahmed Mohammed, Dr. Hafiz Muhammad Faisal Shehzad, and Mr. Ayodeji Ireti Fasiku, for their friendship and for treating me as a brother. I would like to thank my former colleague at American International University-Bangladesh, Mr. Kazi Ahmed Asif Fuad, for his insights regarding my work.

I would like to express my gratitude to my wife, Nusrat Mohaimen, who stood beside me and supported me during all the trials and tribulations of this journey. I could not have completed or even pursued this journey without her support and sacrifice. I would like to thank my sister for her support during tough times throughout this Ph.D. journey. I would like to thank my parents, who encouraged me to take up this endeavor. I would like to express my sincere thanks to everyone who extended his or her support and friendship toward me and my wife during this journey.

ABSTRACT

Conventional convolutional neural networks (CNNs), which are realized in the spatial domain, present a high computational workload and memory access cost (CMC). Spectral domain CNNs (SpCNNs) offer a computationally efficient approach for performing CNN training and inference. State-of-the-art SpCNNs propose activation functions (AFs) that are computationally costly or realize AFs in the spatial domain necessitating multiple and expensive spatial-spectral domain transformations. This work proposes a complex-valued AF for SpCNNs that transmits inputs unaltered or scaled depending on the activation area. This AF is computationally inexpensive and provides sufficient non-linear transformation that ensures high classification accuracy. This work also investigates the CMC of SpCNNs and its contributing components analytically and then proposes a methodology to optimize CMC, under three strategies, to enhance inference and training performance. The strategies involve the reduction of the output feature map (OFM) size, OFM depth, or both under an accuracy constraint to compute performance-optimized CNN inference and training. The proposed AF denoted as complex-valued leaky ReLU (CLReLU), was employed in a LeNet-5 SpCNN architecture and achieves an accuracy gain of up to 3% for MNIST and 8% for Fashion MNIST dataset, while providing up to 2.3 times higher throughput in inference, over state-of-the-art AFs applied to the same model. The proposed CMC reduction methodology was applied to LeNet-5 and AlexNet architectures. For instance, the optimal AlexNet model achieves up to 34 times higher throughput in inference, and up to 16 times greater energy efficiency in training, with a minor accuracy loss of 2%, as compared to related state-of-the-art work. The proposed AF and CMC reduction methodology helps develop an SpCNN model that provides faster and more energy-efficient computation as well as high test accuracy.

ABSTRAK

Rangkaian neural konvolusi (CNNs) konvensional, yang direalisasikan dalam domain spatial, memberikan beban kerja pengiraan dan kos akses memori yang tinggi (CMC). CNN domain spektrum (SpCNNs) menawarkan pendekatan yang cekap dari segi pengiraan untuk melaksanakan latihan dan inferens CNN. SpCNN termaju mencadangkan fungsi pengaktifan (AFs) yang secara pengiraan sangat mahal atau merealisasikan AF dalam domain spatial yang memerlukan transformasi domain spatial-spektrum yang banyak dan komputasi yang tinggi. Kerja ini mencadangkan AF bernilai kompleks untuk SpCNN yang menghantar input sebenar atau berskala bergantung kepada kawasan pengaktifan. AF ini adalah kecil dari segi beban pengiraan dan menyediakan ketaklinearan transformasi yang mencukupi untuk memastikan ketepatan pengelasan yang tinggi. Kerja ini juga menyiasat CMC SpCNN dan komponen penyumbangnya secara analitikal dan kemudian mencadangkan metodologi untuk mengoptimumkan CMC, di bawah tiga strategi, untuk meningkatkan prestasi inferens dan latihan. Strategi tersebut melibatkan pengurangan saiz peta ciri output (OFM), kedalaman OFM, atau kedua-duanya sekali di bawah kekangan ketepatan untuk mengira inferens dan latihan CNN dengan prestasi yang optimum. AF yang dicadangkan dinamakan sebagai ReLU bocor bernilai kompleks (CLReLU), telah digunakan dalam seni bina LeNet-5 SpCNN dan mencapai peningkatan ketepatan sehingga 3% untuk MNIST dan 8% untuk dataset MNIST Fesyen, sambil menyediakan sehingga 2.3 kali daya pemprosesan yang lebih tinggi dalam inferens, ke atas AF termaju yang digunakan pada model yang sama. Metodologi pengurangan CMC yang dicadangkan telah digunakan untuk seni bina LeNet-5 dan AlexNet. Sebagai contoh, model AlexNet yang optimum mencapai sehingga 34 kali daya pemprosesan yang lebih tinggi dalam inferens, dan sehingga 16 kali kecekapan tenaga yang lebih besar dalam latihan, dengan kehilangan ketepatan kecil hanya sebanyak 2%, berbanding dengan kerja terkini yang berkaitan. Metodologi pengurangan AF dan CMC yang dicadangkan membantu membangunkan model SpCNN yang menghasilkan pengiraan yang lebih pantas dan lebih cekap tenaga serta ketepatan yang tinggi.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

DNN	-	deep neural network
CNN	-	convolutional neural network
CONV	-	convolution
CMC	-	computational workload and memory access cost
AI	-	Artificial intelligence
CW	-	computational workload
SpCNN	-	spectral domain CNN
AF	-	activation function
MemAC	-	memory access cost
POOL	-	pooling
FC	-	fully-connected
IFM	-	input feature map
OFM	-	output feature map
CC	-	computational complexity
MB	-	megabyte
DRAM	-	dynamic random access memory
EWM	-	element-wise multiplication
FFT	-	fast Fourier transform
LTI	-	linear time invariant
IFFT	-	inverse FFT
SPOOL	-	spectral pooling
FE	-	feature-extraction
MLP	-	multi-layer perceptron
SMAX	-	softmax
MAC	-	multiply-accumulate
RMAC	-	real-valued multiply-accumulate

ReLU	-	rectified linear unit
ANN	-	artificial neural network
ACReLU	-	auto-convolution-based ReLU
SReLU	-	spectral ReLU
CVCNN	-	complex-valued CNN
SAR	-	synthetic aperture radar
fMRI	-	functional magnetic resonance imaging
LReLU	-	leaky ReLU
CReLU	-	complex ReLU
CCardiode	-	complex cardiode
2SReLU	-	second harmonics SReLU
FReLU	-	Fourier ReLU
FLOP	-	floating-point operation
CFLOP	-	complex-valued FLOP
CMAC	-	complex-valued MAC
OaA	-	overlap-and-add
SQNR	-	signal-to-quantization-noise-ratio
cl	-	number of classifications
cl/s	-	number of classifications per scond
cl/J	-	number of classifications per Joule
W	-	Watt
SGD	-	Stochastic gradient descent
CLReLU	-	complex LReLU
UAA	-	unscaled activation area
SAA	-	scaled activation area
CPA	-	CONV-POOL-AF
DROP	-	dropout
CAP	-	CONV-AF-POOL
RFLOP	-	real-valued FLOP

CA	-	CONV-AF
RCSD	-	relative contribution of OFM size and depth
S1	-	Strategy 1
S2	-	Strategy 2
S3	-	Strategy 3
#CFLOPs	-	Number of CFLOPs

LIST OF SYMBOLS

$F \times F$	-	height and width of IFMs, kernels and OFMs of the CONV layers in the spectral domain
$M \times M$	-	height and width of OFMs of the CONV layers in the spatial domain
$k \times k$	-	height and width of kernels for the CONV layers in the spatial domain
$P \times P$	-	height and width of OFMs for the POOL layers
λ	-	CLReLU scaling factor
x	-	IFM in the spatial domain
w	-	Kernel in the spatial domain
y	-	OFM in the spatial domain
X	-	IFM in the spectral domain
W	-	Kernels in the spectral domain
Y	-	OFM in the spectral domain
$\frac{\partial z}{\partial x}$	-	$IFM - Loss$
$\frac{\partial z}{\partial y}$	-	$OFM - Loss$
$\frac{\partial z}{\partial w}$	-	$Kernel - Loss$
$\frac{\partial y}{\partial x}$	-	partial derivative of output
$F_l \times F_l$	-	height and width of OFMs of the CONV layer l in the spectral domain
D_i	-	IFM depth
D_o	-	OFM depth
α_l	-	shrinkage factor for OFM sizes of the CONV layer l
μ_l	-	growth factor for OFM depths of the CONV layer l
β_m	-	Scaling factor for OFM sizes

β_d	-	Scaling factor for OFM depths
l	-	index of a CONV layer
l_{CV}	-	number of CONV layers
l_{FC}	-	number of FC layers
$D_{i(l)}$	-	IFM depth of the CONV layer l
$D_{o(l)}$	-	OFM depth of the CONV layer l
CW_{FP}	-	CW for forward propagation
CW_{BP}	-	CW for backward propagation
CW_{TRN}	-	CW for training
$CW_{BP-IFM-Loss}$	-	CW for $IFM - Loss$ in backpropagation
$CW_{BP-Kern-Loss}$	-	CW for $Kern - Loss$ in backpropagation
$MemAC_{FP}$	-	MemAC for forward propagation
$MemAC_{BP}$	-	MemAC for backward propagation
$MemAC_{TRN}$	-	MemAC for training
$MemAC_{BP-IFM-Loss}$	-	MemAC for $IFM - Loss$ in backpropagation
$MemAC_{BP-Kern-Loss}$	-	MemAC for $Kernel - Loss$ in backpropagation

CHAPTER 1

INTRODUCTION

Deep neural networks (DNNs) have recently evolved as the prevalent solution for a range of challenging problems in computer vision [1], language processing [2] and autonomous systems [3]. Convolutional neural networks (CNNs) [4, 5], a class of DNNs, have achieved unprecedented success in various fields of computer vision, audio analysis, and text processing including–inter alia–object classification [6, 7], object detection [8, 9], semantic segmentation [10, 11], face verification [12, 13], video understanding [14], audio classification [15, 16], and natural language processing [17]. In the last few years, CNNs have been deployed in diverse applications such as autonomous driving [18], navigation systems [19] for drones, skin cancer detection [20], and VLSI physical design [21].

CNNs possess excellent recognition capabilities because they can autonomously extract complex and sophisticated features from incoming data through their convolution (CONV) layers. However, because the CONV layers exhibit high computational workload and memory access cost (CMC), CNN-based artificial intelligence (AI) solutions are power-hungry and inefficient in terms of throughput and energy efficiency [22]. This becomes a concern for the applications discussed in the previous paragraph.

CNN can be computed in the spectral domain to lower the computational workload (CW) of conventional CNN realizations. However, spectral domain CNNs (SpCNN) lack activation functions (AF) that are computationally inexpensive and exhibit sufficient non-linearity to provide high accuracy [23]. Additionally, few studies have looked into approaches to lower memory access costs (MemACs) for SpCNNs. Furthermore, for systems with limited resources and energy, the CW reduction might not be enough. Therefore, it is essential that AFs for SpCNNs are designed that are computationally inexpensive and exhibit adequate non-linearity, and SpCNN is computed with lower CMC than state-of-the-art SpCNN models.

1.1 Convolutional Neural Network

CNN is a variant of DNNs that is built as a series of CONV layers that are interleaved with AF and pooling (POOL) layers, followed by fully-connected (FC) layers, and ends with an AF, such as *softmax*, that performs multi-class classification. The CONV layers process higher-level data called feature maps [22]. The input and output of a CONV layer are thus called input feature map (IFM) and output feature map (OFM), respectively. Figure 1.1 depicts this high-level functional architecture of CNNs. Conventional CNNs, which are typically realized in the spatial domain, implement CONV operation by repeatedly performing dot-products between a moving patch of IFM and a pre-computed kernel. The moving patch is a local region of IFM, called the receptive field, which has the same size as the kernel. The dot-products are generated from accumulating element-wise products between elements of the receptive field and kernel elements [4–6]. It is worth noting that the kernel values are learned (progressively fine-tuned) by CNN through analyzing large amounts of example data—a process called learning or training. When a CNN processes unseen data–a process called inference–the pre-computed kernel values coming out of training are utilized.

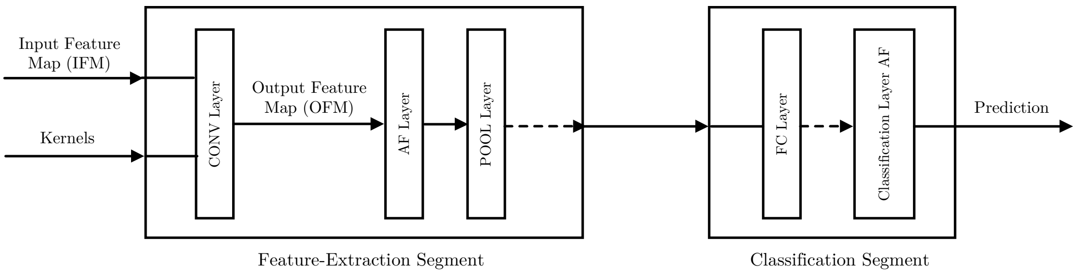


Figure 1.1 High-level functional architecture of CNNs

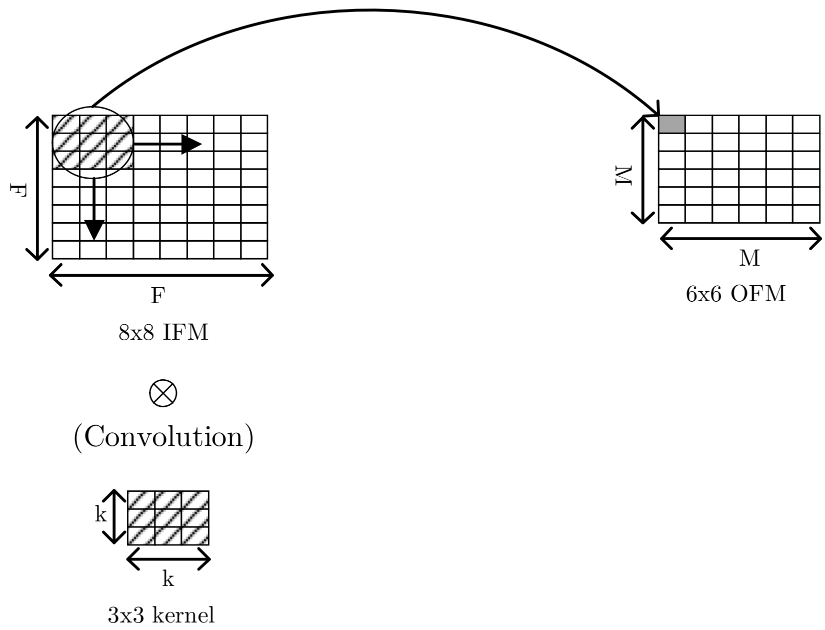
In CNNs, CONV layers play a central role in feature-extraction [4–6] but demand high computational resources [22]. They account for 90% of CNN operations

[24]. The computational complexity (CC) of a CONV operation is $O(M^2k^2)$, where $M \times M$ and $k \times k$ represent the sizes of the OFM and the kernel, respectively [25, 26]. Here, each OFM element produced by a CONV layer requires k^2 multiply-accumulate operations, or MAC operations in short, as shown in Figure 1.2. Since CONV layers in CNNs process many IFMs and produce even a greater number OFMs (in some cases, hundreds), the actual number of arithmetic operations for even a moderately deep CNN model such as AlexNet [6] can exceed 700 thousand MAC operations [27]. This amounts to 1.4 million arithmetic operations in total as one MAC operation involves two arithmetic operations.

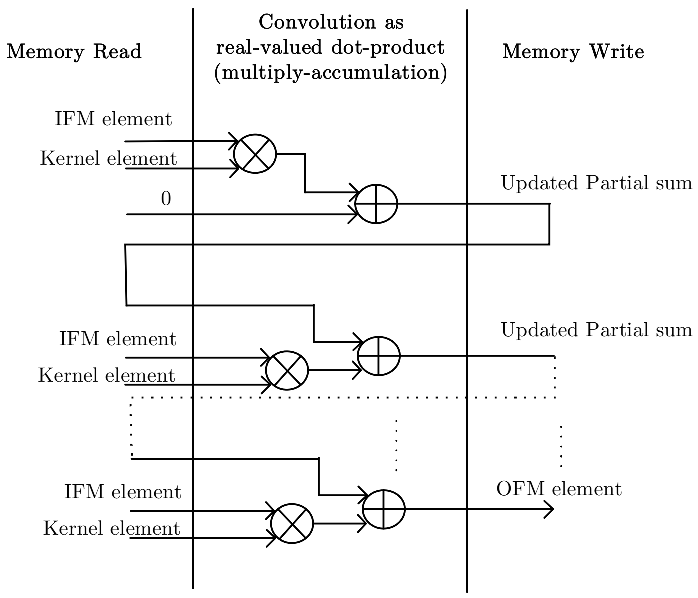
CONV operation in the spatial domain is also expensive in terms of the number of memory accesses required. For a kernel of size $k \times k$, computation of one OFM element requires $4k^2$ memory accesses in total, where reading IFM pixels, kernel elements and partial sums generated during dot-product computations requires k^2 memory accesses each. Another k^2 memory accesses are needed to store either the updated partial sums or the computed OFM element. Therefore, each MAC operation needs three memory-read operations and one memory-write operation [22]. This is illustrated in Figure 1.2.

Computing training and inference for CNNs require storage of kernel values (also known as parameters) and the OFMs of CONV layers (also known as activations) during both deployments, i.e., inference, and offline training [22, 28]. Memory consumed for the former is called parameter memory, while that of the latter is called activation memory. Deeper CNNs (with more CONV layers), which tend to produce higher accuracy, possess a larger number of parameters and activations. This results in increased memory requirements [22]. For example, a deep CNN model such as VGG-19 [29] (containing 19 CONV layers) possess over 140 thousand parameters and hence, requires over 500 megabytes (MB) of parameter memory [27].

Researchers have shown that activation memory dominates the memory footprint during training [30]. In certain scenarios, activation memory can take up more space than parameter memory during inference as well. This can occur when the CNN model is deeper [31] or the inputs are high-resolution images [10]. In some



(a)



(b)

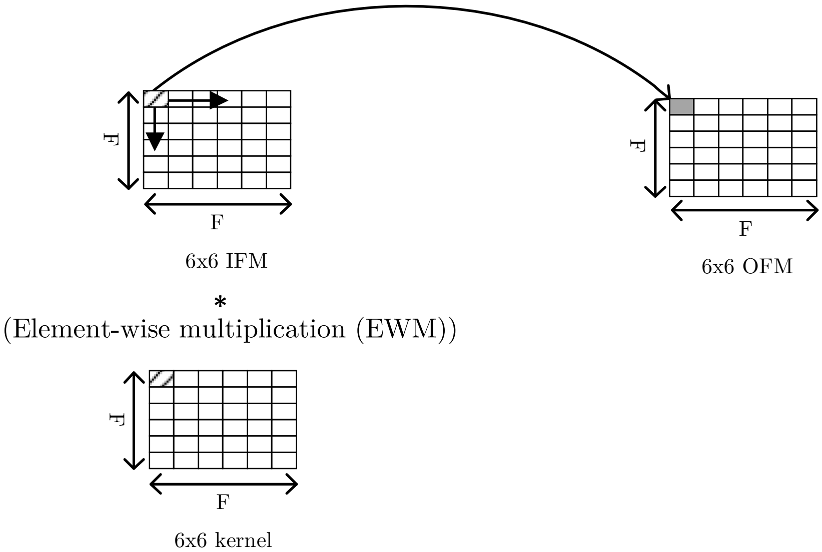
Figure 1.2 (a) CW and (b) MemAC of computing one OFM element in spatial-domain CONV. In spatial domain CONV, IFMs and kernels with sizes $F \times F$ and k×k, respectively, produce OFMs with a size $M \times M$, where $M = F - (k - 1)$. For instance, an 8×8 IFM convolved with a 3×3 kernel produces a 6×6 OFM.

of these cases, activation memory can be ten to a hundred times larger than parameter memory [10]. Therefore, conducting training and, in some cases, inference necessitates the storage of parameters or activations or both in off-chip memory devices such as dynamic random access memory (DRAM). However, accessing data from DRAMs consumes more power and energy than computations [22, 32]. In fact, for devices limited by memory bandwidth, the MemAC can be the main bottleneck for power consumption and inference latency including in GPU-based platforms [10, 33, 34].

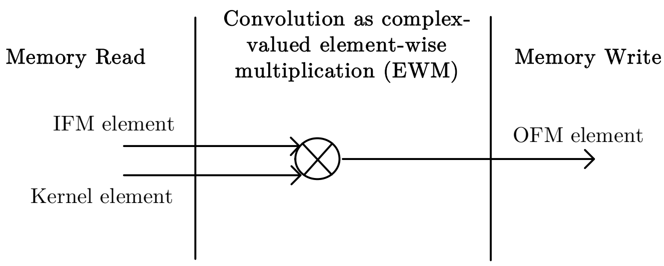
1.2 Spectral Domain Convolutional Neural Network

An effective approach to reduce the CW of conventional CNNs is to represent CNNs, especially CONV, in spectral domain using Fourier transformation [25, 26, 35, 36]. Furthermore, SpCNNs can achieve high accuracy in character and face recognition [23]. It is worth noting that the word ”spectral domain” in this work refers to the frequency domain, where data is transformed from the spatial domain to the spectral domain using Fourier transformation. Spectral domain CNNs (SpCNNs) compute CONV operation as element-wise multiplication (EWM) in Fourier space, which significantly reduces the CW. Here, each OFM element can be computed from one complex-valued product, instead of many real-valued products accumulated over the receptive field, as is the case in spatial domain [25, 26, 35, 36]. This is illustrated in Figure 1.3. CONV operation realized as EWM in spectral domain exhibits a CC of $O(F^2)$, where $F \times F$ represents the size of the OFM, which is computationally much lighter than the $O(M^2k^2)$ complexity of spatial-domain CONV operation [25].

SpCNNs require a comparatively lesser number of memory accesses for computing OFM elements through EWM as compared to spatial-domain CONV computation since CONV operation in an SpCNN is computed in an element-wise manner. Instead of $4k^2$ memory accesses required for computing one OFM element in spatial-domain, EWM can compute the same OFM element with just three memory accesses [22]. Here, reading the IFM element and the kernel element requires one memory access each, while one memory access is needed for writing the computed OFM element to memory. Even though the complex-valued product required to



(a)



(b)

Figure 1.3 (a) CW and (b) MemAC of computing one OFM element in spectral-domain EWM. In EWM, both IFMs and kernels have a size $F \times F$, and EWM between them produces OFMs with a size $F \times F$. For instance, EWM of a 6×6 IFM and a 6×6 kernel produces a 6×6 OFM.

compute each OFM element involves 4 real-valued products and two real-valued additions [25], the computations do not involve any accumulation with previously computed sums. Therefore, intermediate results do not read to be stored in or read from memory. This is illustrated in Figure 1.3.

1.3 Problem Statement

One of the key issues in spectral domain CNNs is the lack of activation functions (AFs) that are effective in the spectral domain [35]. AFs provide non-linearity to CONV layers so that they can extract complex features. The non-linear nature of AFs is incompatible with the linear time-invariant (LTI) property of spectral domain [23]. Therefore, early SpCNNs computed only CONV operation in spectral domain [25, 26] as non-linear layers such as pooling (POOL), employed for dimensionality reduction, and AF were not formulated yet. This required domain transformations before and after each CONV layer (using FFT and inverse FFT (IFFT)) so that CONV layers can perform computations in the spectral domain, while non-linear layers can do so in the spatial domain. This architecture is depicted in Figure 1.4. These multiple domain transformations, with a CC of $O(F^2 log_2 F)$ [25], are computationally expensive. The computational cost of these transformations negated some of the gains in computational efficiency achieved with SpCNNs [23, 36]. Even after spectral POOL (SPOOL) operation was developed by Rippel *et al.* [35] multiple domain transformations could not be avoided as AFs were still computed in the spatial domain. Previous attempts at spectral domain AF had the drawbacks of insufficient non-linearity [37] or exhibited high CC [23]. It is worth noting that recently proposed AFs (such as the one proposed by [23]) require CONV operation to compute them. This goes against the rationale for computing CNN in the spectral domain in the first place, which is to replace computationally costly CONV operation with computationally inexpensive EWM. Therefore, computing AF in the native spectral domain and in a computationally inexpensive manner is an important problem in the field of SpCNN.

As discussed previously, SpCNNs can reduce CW significantly as compared to conventional spatial-domain approaches. However, certain scenarios may demand

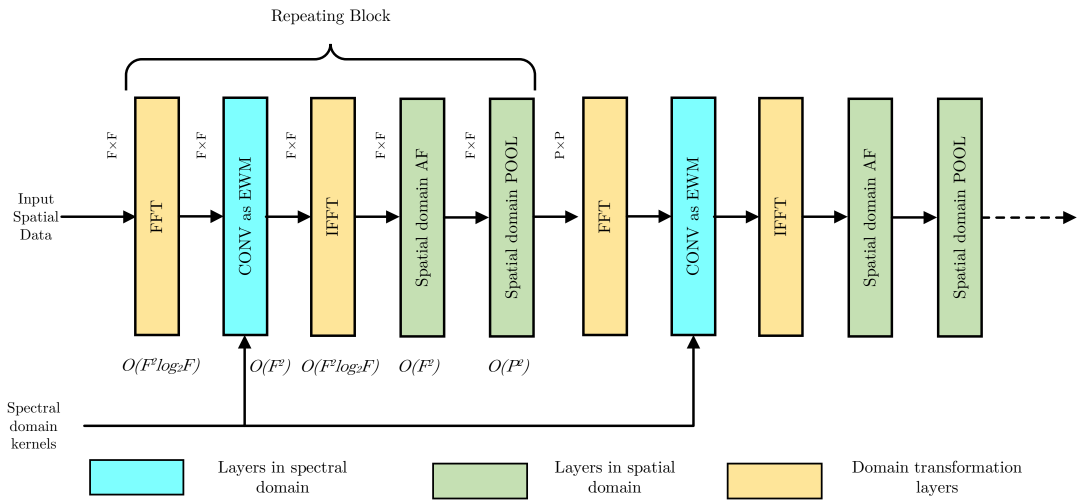


Figure 1.4 Domain transformations in conventional SpCNNs. The POOL layers reduce OFM dimensions from $F \times F$ to $P \times P$, where $P < F$.

further reduction in CW For instance, the reduction of CW for SpCNNs becomes important for deeper models and resource-constrained platforms. Existing approaches that attempt to further reduce the CW of SpCNNs include a quantization approach [38] to represent data (kernels, IFMs, OFMs) with reduced precision and the use of fused CONV layers [23]. The quantization approaches are tailored for specific implementation platforms and hence, require a dedicated hardware accelerator to take advantage of these approaches. In fused CONV, POOL operation is performed before CONV, so that CONV layers process smaller-sized IFMs and consequently produce smaller-sized OFMs. This approach optimizes OFM sizes across all CONV layers without considering where the size reduction (which CONV layer) would be impactful. Furthermore, works that employ these approaches do not consider optimizing OFM depths, which is a significant contributor to both CW and MemAC. Therefore, an analytical formulation for CW and an associated optimization methodology are necessary for identifying factors affecting CW and where, i.e., in which CONV layer, optimizing it would be most effective.

A feature of SpCNNs is that they require larger-sized kernels as compared to spatial-domain CNNs [23]. Because of the element-wise nature of EWM computation, kernels in SpCNNs have to be the same size as IFMs. As a result, a CNN architecture implemented as an SpCNN has a larger parameter memory than a spatial-domain implementation. For instance, it can be shown that an SpCNN realization of LeNet-5 or AlexNet architectures requires twice the number of parameters and hence a two-fold larger parameter memory as compared to their spatial-domain counterparts. It is worth noting that kernels in SpCNNs are typically computed in the spatial domain during the training phase [23, 25]. So the pre-computed kernels are transformed from the spatial domain to the spectral domain through fast Fourier transform (FFT) first. Then kernels for each CONV layer are resized to match the size of IFMs of that layer before providing them to the CNN for computing inference [23].

A few previous studies have explored methods to reduce memory usage in SpCNNs. Many of these approaches aim to reduce either the size of parameter memory by pruning [39] or quantizing kernels [38] or reducing the memory cost of domain transformations [24]. These works do not address the reduction of activation memory. In addition, these works require a dedicated hardware accelerator to take advantage of these approaches. One work authored by Guan *et al.* [40] investigates the reduction of activation memory for faster training by compressing IFMs for sparse storage of activations (OFMs produced by CONV layers). Approaches such as [40] do not reduce the number of memory accesses, and generally requires specialized libraries designed for sparse matrices to take advantage of such approaches. It is worth noting that the reduction of the number of memory accesses is critical for reducing power consumption and improving the energy efficiency of any CNN model [22, 32]. However, the existing approaches do not analytically investigate factors that contribute to MemAC or offer a structured strategy to optimize this cost.

1.4 Research Objectives

This thesis aims to develop an SpCNN model that employs a computationally efficient AF with sufficient non-linear characteristics and that can be optimized to produce faster and more energy-efficient training and inference with smaller memory footprint as compared to state-of-the-art SpCNN models. The objectives of this thesis are outlined below.

1. To propose an AF for SpCNNs that can be computed in the native spectral domain, which is computationally efficient and has sufficient non-linearity to help achieve high classification accuracy when deployed in an SpCNN model.
2. To propose a baseline SpCNN model employing the proposed AF that offers accurate and faster inference over SpCNN models that employ state-of-the-art spectral-domain AFs.
3. To develop an analytical model for the CW and MemAC of SpCNNs to help designers analyze and improve the computational and memory efficiency of SpCNN models.
4. To propose an optimization methodology, based on the analytical model for the CW and MemAC, that can optimize a baseline SpCNN model for higher computational efficiency and smaller memory footprint under an accuracy constraint.

1.5 Scope of Work

Research works on CNNs, including work on SpCNNs, have been conducted on various aspects of CNN computations, applications, and deployment. The scope of this work is outlined below.

1. The SpCNN models were evaluated on LeNet-5 and AlexNet CNN architectures. These architectures are ideal for validating algorithms and optimization strategies for higher computational and memory efficiency as they have computationally expensive and parameter-heavy CONV layers. Therefore, architectures with 1×1 CONV operations such as MobileNet [41] are not considered for this work.
2. The effectiveness of the proposed methodology and SpCNN models are demonstrated on the MNIST dataset [42] for handwritten digits and the Fashion MNIST dataset [43] for fashion articles. These are two well-established image classification datasets that are widely used in the deep learning research

community for exploring and benchmarking new models, methodologies, and algorithms.

3. The accuracy of SpCNN models is measured in terms of the total amount of correct classifications conducted during inference in percentages, while throughput in training and inference are measured in terms of the number of classifications performed per second on the training and test set, respectively. Power consumption is measured in Watt (Joule per second) and energy efficiency in training and inference are measured in terms of the number of classifications conducted on the training and test set, respectively, per unit amount of energy, i.e., Joule. Memory consumption is measured in terms of the amount of off-chip memory consumed in MB. These are standard performance metrics for analyzing the performances of a CNN model in training and inference.

4. All the SpCNN models developed for this work are trained and tested in MATLAB computing platform, developed by MathWorks, and utilizing the MatConvNet deep learning library, developed by Vedaldi *et al.* [44] at Visual Geometry Group of the University of Oxford.

1.6 Research Contributions

This work develops a computationally efficient AF that can be computed in the native spectral domain, a baseline SpCNN model employing this AF that exhibits fast and accurate inference, an analytical model for CMC for SpCNNs, and an optimization methodology for SpCNN models to compute high-throughput and energy-efficient training and inference with only a minor loss in accuracy.

The major contributions of this work are summarized below.

1. The proposed AF exhibits lower CC than state-of-the-art spectral-domain AFs and exhibit high non-linearity. It has a CC of $O(F^2)$ and can be computed without requiring any computationally-intensive operations (e.g., multiplication or CONV). The AF can be computed in the native spectral domain and

hence, removes the need for computationally expensive multiple sets of domain transformations when employed in an SpCNN model.

2. The proposed baseline SpCNN model employing the proposed AF exhibits lower CC and sufficient non-linearity to achieve up to 2.3× higher throughput and up to 8% higher classification accuracy than the same model employing state-of-the-art spectral-domain AFs.

3. The proposed analytical formulations for the CW and MemAC allow designers to identify structural hyper-parameters that contribute more to the CW and MemAC as well as how these affect different CONV layers. In addition, the analytical formulations allow the CW and MemAC to be estimated before conducting training or inference.

4. The proposed methodology, based on the analytical formulations, develops a design flow for optimizing two structural hyper-parameters of an SpCNN model, the OFM sizes and depths, under three strategies. The strategies involve the reduction of the OFM size, OFM depth, or both, to compute performance-optimized CNN inference and training under an accuracy constraint. When the proposed methodology is applied to LeNet-5 SpCNN, the optimal model achieves up to 5× higher throughput during inference, and up to 4× greater energy efficiency in training with a maximum loss of accuracy of just 3%, when compared to the state-of-the-art SpCNN models. The proposed methodology is algorithmic and hence, does not require a dedicated accelerator or a specialized module.

1.7 Thesis Organization

The remainder of the thesis is structured in the following manner. Chapter 2 introduces CNNs and SpCNNs, and reviews previous studies conducted on AFs for SpCNNs and CMC reduction. Chapter 3 discusses architectures, datasets, design tools and experimental environments for evaluating the SpCNN models. It also discusses training and inference methodology as well as performance metrics and accuracy constraint for the SpCNN models. Chapter 4 discusses the mathematical formulation of the proposed AF and introduces the proposed baseline LeNet-5 SpCNN model to evaluate this AF. Chapter 5 presents an analytical model for the CMC of SpCNNs and the proposed methodology for reducing the CMC to enhance inference and training performance. This chapter also introduces the baseline LeNet-5 and AlexNet SpCNN models for evaluating the proposed methodology. Afterward, Chapter 6 discusses the experimental methodology and results. In the end, Chapter 7 presents the concluding remarks.

REFERENCES

1. Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., AlShamma, O., Santamar´ıa, J., Fadhel, M.A., Al-Amidie, M. and Farhan, L. Review of deep learning-concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 2021. 8(1): 1–74.

2. Otter, D. W., Medina, J. R. and Kalita, J. K. A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2021. 32(2): 604–624.

3. Grigorescu, S., Trasnea, B., Cocias, T. and Macesanu, G. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 2020. 37(3): 362–386.

4. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W. and Jackel, L. Handwritten Digit Recognition with a Back-Propagation Network. *Proceedings of the 2nd International Conference on Neural Information Processing Systems (NIPS)*. Denver, CO, USA. 1989. 396—404.

5. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 86(11): 2278–2324.

6. Krizhevsky, A., Sutskever, I. and Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, 2017. 60(6): 84–90.

7. Hu, J., Shen, L., Albanie, S., Sun, G. and Wu, E. Squeeze-and-Excitation Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 42(8): 2011–2023.

8. Cao, C., Wang, B., Zhang, W., Zeng, X., Yan, X., Feng, Z., Liu, Y. and Wu, Z. An improved faster R-CNN for small object detection. *IEEE Access*, 2019. 7: 106838—106846.

9. Aziz, L., Haji Salam, M.S.B., Sheikh, U.U. and Ayub, S. Exploring deep learning-based architecture, strategies, applications and current trends in

generic object detection: A comprehensive review. *IEEE Access*, 2020. 8: 170461—170495.

10. Shelhamer, E., Long, J. and Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017. 39(4): 640—651.

11. Li, C., Xia, W., Yan, Y., Luo, B. and Tang, J. Segmenting objects in day and night: Edge-conditioned CNN for thermal image semantic segmentation. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2021. 32(7): 3069—3082.

12. Kang, S., Lee, J., Bong, K., Kim, C., Kim, Y. and Yoo, H.-J. Low-power scalable 3-d face frontalization processor for CNN-based face recognition in mobile devices. *IEEE Journal on Emerging and Selected Topics In Circuits and Systems (JETCAS)*, 2018. 8(4): 873—883.

13. Jiang, L., Zhang, J. and Deng, B. Robust RGB-D face recognition using attribute-aware loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 42(10): 2552—2566.

14. Khurana, K., Deshpande, U. Video question-answering techniques, benchmark datasets and evaluation metrics leveraging video captioning: A comprehensive survey. *IEEE Access*, 2021. 9:43799—43823.

15. Lin, Y., Guo, D., Zhang, J., Chen, Z. and Yang, B. A unified framework for multilingual speech recognition in air traffic control systems. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2021. 32(8): 3608–3620.

16. Kim, T., Lee, J. and Nam, J. Comparison and analysis of sample CNN architectures for audio classification. *IEEE Journal of Selected Topics In Signal Processing (JSTSP)*, 2019. 13(2): 285–297.

17. Ramisa, A., Moreno-Noguer, F. and Moreno-Noguer, K. BreakingNews: Article annotation by image and text processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018. 40(5): 1072–1085.

18. Chen, L., Lin, S., Lu, X., Cao, D., Wu, H., Guo, C., Liu, C. and Wang, F.-Y. Deep neural network based vehicle and pedestrian detection for autonomous

driving: A survey. *IEEE Transactions On Intelligent Transportation Systems (TITS)*, 2021. 22(6): 3234—3246.

19. Miclea, V.-C. and Nedevschi, S. Monocular depth estimation with improved long-range accuracy for UAV environment perception. *IEEE Transactions On Geoscience and Remote Sensing (TGRS)*, 2022. 60: 1—15.

20. Esteva, A., Kuprel, B., Novoa, R., Ko, J., Swetter, S., Blau, H. and Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 2017. 542(7639): 115–118.

21. Saraogi, E., Chouhan, G., Panchal, D., Patel, M. and Gajjar, R. CNN based design rule checker for VLSI layouts. *Proceedings of the 2nd IEEE International Conference on Applied Electromagnetics, Signal Processing & Communication (AESPC)*. Bhubaneswar, India. 2021. 1—6.

22. Sze, V., Chen, Y.-H., Yang, T.-J. and Emer, J. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE*, 2017. 105(12): 2295–2329.

23. Ayat, S., Khalil-Hani, M., Ab Rahman, A. and Abdellatef, H. Spectral-based convolutional neural network without multiple spatial-frequency domain switchings. *Neurocomputing*, 2019. 364: 152–167.

24. Abtahi, T., Shea, C., Kulkarni, A., Mohsenin, T. Accelerating convolutional neural network with FFT on embedded hardware. *IEEE Transactions on Very Large Scale Integration Systems (TVLSI)*, 2018. 26(9): 1737–1749.

25. Mathieu, M., Henaff, M. and LeCun, Y. Fast training of convolutional networks through FFTs. *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*. Banff, AB, Canada. 2014. 1–9.

26. Vasilache, N., Johnson, J., Mathieu, M., Chintala, S., Piantino, S. and LeCun, Y. Fast convolutional nets with fbfft: A GPU performance evaluation. *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. San Diego, CA, USA. 2015. 1–17.

27. Abdelouahab, K., Pelcat, M. and Berry, F. Accelerating the CNN inference on FPGAs. In: Fagerberg, J., Mowery, D.C. and Nelson, R.R., eds. *Deep Learning*

in Computer Vision: Principles and Applications. CRC Press Taylor & Francis Group. 2020. 1–39.

28. Liu, Z., Li, J., Shen, Z., Huang, G., Yan, S. and Zhang, C. Learning efficient convolutional networks through network slimming. *Proceedings of the 16th IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy. 2017. 2755—2763.

29. Simonyan, K. and Zisserman, A. Very Deep Convolutional Networks for LargeScale Image Recognition. *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. San Diego, CA, USA. 2015. 1–14.

30. Jain, A., Phanishayee, A., Mars, J., Tang, L., Pekhimenko, G. Gist: Efficient data encoding for deep neural network training. *Proceedings of the 45th International Symposium on Computer Architecture (ISCA)*. Los Angeles, CA, USA. 2018. 776—789.

31. Minakova, S. and Stefanov, T. Buffer Sizes Reduction for Memory-efficient CNN Inference on Mobile and Embedded Devices. *Proceedings of the 23rd Euromicro Conference on Digital System Design (DSD)*. Kranj, Slovenia. 2020. 133–140.

32. Chen, Y.-H., Krishna, T., Emer, J.S. and Sze, V. Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. *IEEE Journal of Solid-State Circuits (JSSC)*, 2017. 52(1): 127–138.

33. Ma, N., Zhang, X., Zheng, H.-T., Sun, J. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. *Proceedings of the 15th European Conference on Computer Vision (ECCV)*. Munich, Germany. 2018. 116–131.

34. Vaze, S., Xie, W. and Namburete, A. I. Low-memory CNNs enabling realtime ultrasound segmentation towards mobile deployments. *IEEE Journal of Biomedical and Health Informatics (JBHI)*, 2020. 24(4): 1059–1069.

35. Rippel, O., Snoek, J. and Adams, R. Spectral representations for convolutional neural networks. *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)*. Montréal, QC, Canada. 2015. 2449—2457.

36. Ko, J., Mudassar, B., Na, T. and Mukhopadhyay, S. Design of an energy-efficient accelerator for training of convolutional neural networks using frequency-domain computation. *Proceedings of the 54th ACM/EDAC/IEEE Design Automation Conference (DAC)*. Austin, TX, USA. 2017. 1—6.

37. Arjovsky, M., Giguere, P. and Bengio, Y. Unitary evolution recurrent neural networks. *Proceedings of the 33rd International Conference on Machine Learning (ICML)*. New York City, NY, USA. 2016. 1120–1128.

38. Sun, W., Zeng, H., Yang, Y.-h. and Prasanna, V. Throughput-optimized frequency domain CNN with fixed-point quantization on FPGA. *Proceedings of the 13th International Conference on ReConFigurable Computing and FPGAs (ReConFig)*. Cancun, Mexico. 2018. 1—8.

39. Niu, Y., Zeng, H., Srivastava, A., Lakhotia, K., Kannan, R., Wang, Y. and Prasanna, V. SPEC2: SPECtral SParsE CNN accelerator on FPGAs. *Proceedings of the 26th IEEE International Conference on High Performance Computing, Data, and Analytics (HiPC)*. Hyderabad, India. 2019. 195—204.

40. Guan, B., Zhang, J., Sethares, W., Kijowski, R. and Liu, F. Spectral domain convolutional neural network. *Proceedings of the 46th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Toronto, ON, Canada. 2021. 2795–2799.

41. Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

42. Deng, L. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. *IEEE Signal Processing Magazine*, 2012. 29(6): 141-142.

43. Xiao, H., Rasul, K., Vollgraf, R. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

44. Vedaldi, A., Lux, M. and Bertini, M. MatConvNet: CNNs are also for MATLAB users. *ACM SIGMultimedia Records*, 2018. 10(1): 9—9.

45. Buduma, N. and Locascio., N. *Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms*. O'Reilly Media, Inc. 2017.

46. Aghdam, H. and Heravi, E. *Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification*. Springer International Publishing. 2017.

47. M. Sipser, *Introduction To The Theory of Computation*, 3rd Edition. Cengage Learning. 2013.

48. Oppenheim, A. and Schafer, R. *Discrete-Time Signal Processing*. Pearson Education. 2014.

49. Wang, Z., Lan, Q., Huang, D. and Wen, M. Combining FFT and Spectral-Pooling for Efficient Convolution Neural Network Model. *Proceedings of the 2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE)*. Beijing, China. 2016. 203–206.

50. Apicella, A., Donnarumma, F., Isgrò, F. and Prevete, R. A survey on modern trainable activation functions. *Neural Networks*, 2021. 138: 14–32.

51. Lau, M. M. and Lim, K. H. Review of Adaptive Activation Function in Deep Neural Network. *Proceedings of the 5th IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*. Sarawak, Malaysia. 2018. 686–690.

52. Wang, Y., Li, Y., Song, Y., and Rong, X. The influence of the activation function in a convolution neural network model of facial expression recognition. *Applied Sciences*, 2020. 10(5): 1–20.

53. Agostinelli, F., Hoffman, M., Sadowski, P. and Baldi, P. Learning activation functions to improve deep neural networks. *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. San Diego, CA, USA. 2015. 1–9.

54. Kuo, C.-C. J. Understanding convolutional neural networks with a mathematical model. *Journal of Visual Communication and Image Representation*, 2016. 41: 406–413.

55. Lee, W.-Y., Park, S.-M. and Sim, K.-B. Optimal hyperparameter tuning of convolutional neural networks based on the parameter-setting-free harmony search algorithm. *Optik*, 2018. 172: 359–367.

56. Uijens, W. *Activating Frequencies-Exploring Non-Linearities in the Fourier Domain*. Master thesis. Delft University of Technology, The Netherlands. 2018.

57. Glorot, X., Bordes, A. and Bengio, Y. Deep Sparse Rectifier Neural Networks. *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*. Reykjavik, Iceland. 2011. 315–323.

58. Scardapane, S., Van Vaerenbergh, S., Hussain, A. and Uncini, A. Complex-valued neural networks with nonparametric activation functions. *IEEE Transactions on Emerging Topics in Computational Intelligence (TETCI)*, 2018. 4(2): 140–150.

59. Pan, H., Chen, Y., Niu, X., Zhou, W., Li, D. Learning Convolutional Neural Networks in the Frequency Domain. *arXiv preprint arXiv:2204.06718v3*, 2022.

60. Trottier, L., Shah, A., Chaib-draa, B. Parametric Exponential Linear Unit for Deep Convolutional Neural Networks. *Proceedings of the 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. Cancun, Mexico. 2017. 207–214.

61. Han, Y. and Hong, B.-W. Deep Learning based on Fourier convolutional neural network incorporating random kernels. *Electronics*, 2021. 10(16): 1-19.

62. Virtue, P., Yu, S. and Lustig, M. Better than real: complex-valued neural nets for MRI fingerprinting. *Proceedings of the 24th IEEE international conference on image processing (ICIP)*. Beijing, China. 2017. 3953–3957.

63. Watanabe, T., Wolf, D. Image classification in frequency domain with 2SReLU: a second harmonics superposition activation function. *Applied Soft Computing*, 2021. 112: 107851-107851.

64. Lin, J., Ma, L. and Yao, Y. A Fourier domain acceleration framework for convolutional neural networks. *Neurocomputing*, 2019. 364: 254–268.

65. Chao, P., Kao, C.-Y., Ruan, Y., Huang, C.-H. and Lin, Y.-L. HarDNet: A low memory traffic network. *Proceedings of the 17th IEEE/CVF International*

Conference on Computer Vision (ICCV). Seoul, Korea (South). 2019. 3551–3560.

66. Highlander, T. and A Rodriguez, A. Very Efficient Training of CNNs using Fast Fourier Transform and Overlap-and-Add, *arXiv preprint arXiv:1601.06815*, 2016.

67. Ayat, S. *Accurate, and Energy-Efficient Spectral Convolutional Neural Network Accelerator on Field-Programmable Gate Array*. PhD thesis. Universiti Teknologi Malaysia, Johor Bahru, Malaysia. 2020.

68. Li, G., Xue, J., Liu L, Wang, X., Ma, X., Dong, X., Li, J. and Feng, X. Unleashing the Low-Precision Computation Potential of Tensor Cores on GPUs. *Proceedings of the 19th IEEE/ACM International Symposium on Code Generation and Optimization (CGO)*. Seoul, Korea (South). 2021. 90-102.

69. Rokh, B., Azarpeyvand, A. and Khanteymoori, A. A Comprehensive Survey on Model Quantization for Deep Neural Networks. *arXiv preprint arXiv:2205.07877*, 2021.

70. Gholami, A., Kim, S., Dong, Z., Yao, Z., Mahoney, M. and Keutzer, K. A survey of quantization methods for efficient neural network inference. *arXiv preprint arXiv:2103.13630*, 2021.

71. Dziedzic, A., Paparrizos, J., Krishnan, S., Elmore, A. and Franklin, M. Band-limited training and inference for Convolutional Neural Networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*. Long Beach, CA, USA. 2019. 1745–1754.

72. Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F., Bengio, Y. and Courville, A. On the Spectral Bias of Neural Networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*. Long Beach, CA, USA. 2019. 5301–5310.

73. About MatConvNet. `https://www.vlfeat.org/matconvnet/about/`, accessed February 21, 2023.

74. Vedaldi, A., Lenc, K. Matconvnet: Convolutional neural networks for matlab. *Proceedings of the 23rd ACM International Conference on Multimedia (MM)*. Brisbane, Australia. 2015. 689–692.

75. Wang, Z., Liu, K., Li, J., Zhu, Y., Zhang, Y. Various frameworks and libraries of machine learning and deep learning: a survey. *Archives of Computational Methods in Engineering*, 2019. 2019: 1-24.

76. Andrea Vedaldi. Deep learning research in MATLAB, MATLAB Expo, 2017. `https://www.matlabexpo.com/content/dam/mathworks/mathworks-dot-com/images/events/matlabexpo/uk/2017/matconvnet-deep-learning-research-in-matlab.pdf`, accessed September 30, 2022.

77. Mathworks reference. tanh. `https://www.mathworks.com/help/matlab/ref/tanh.html`, accessed February 21, 2023.

78. Mathworks reference. tanhLayer. `https://www.mathworks.com/help/deeplearning/ref/nnet.cnn.layer.tanhlayer.html`, accessed February 21, 2023.

79. Meurant, G. *Computer Solution of Large Linear Systems*. Elsevier. 1999.

80. Sadouk, L. CNN approaches for time series classification. In: Ngan, C.-K., eds. *Time Series Analysis-Data, Methods, and Applications*. 2019. 57–79.

81. Wang, E., Davis, J., Zhao, R., Ng, H.-C., Niu, X., Luk, W., Cheung, P., Constantinides, G. Deep neural network approximation for custom hardware: Where we've been, where we're going. *ACM Computing Surveys*, 2019. 52(2): 1-39.

82. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A. Going deeper with convolutions. *Proceedings of the 28th IEEE conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA, USA. 2015. 1–9.

83. He, K., Zhang, X., Ren, S. and Sun, J. Deep residual learning for image recognition. *Proceedings of the 29th IEEE conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA. 2016. 770–778.

LIST OF PUBLICATIONS

Journal with Impact Factor

1. Rizvi, S., Ab Rahman, A., Sheikh, U., Fuad, K., and Shehzad, H. Computation and memory optimized spectral domain convolutional neural network for throughput and energy-efficient inference. *Applied Intelligence*, 2023. 53:4499-4523. (ISI, IF: 5.019)

Other Journal Publications

1. Rizvi, S., Ab Rahman, A., Khalil-Hani, M. and Ayat, S. A Low-complexity Complex-valued Activation Function for Fast and Accurate Spectral Domain Convolutional Neural Network. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, 2021. 9(1): 173-184. (Scopus, Citescore: 1.5)

Indexed Conference Proceedings

1. Ayat, S., Rizvi, S., Abdellatef, H., Ab Rahman, A., and Manan, S. A Compact Spectral Model for Convolutional Neural Network. *Proceedings of the 2022 Future Technologies Conference (FTC), Volume 1.* Vancouver, Canada. 2023. 100-120. *Lecture Notes in Networks and Systems*, 2023. 559. (Scopus-indexed)