ENHANCING CNN MODELS WITH WAVELET SCATTERING TRANSFORM FOR IMPROVED FEATURE EXTRACTION AND TRANSLATION INVARIANCE

Khaled Almutairy[1], *Mamoun Alghaslan*[1], *Saeed Anwar*[1,2]

SDAIA-KFUPM Joint Research Center for Artificial Intelligence[1]
Information and Computer Science, King Fahd University of Petroleum and Minerals, Saudi Arabia[2]
{g202204240, g200818720, saeed.anwar}@kfupm.edu.sa

ABSTRACT

Building an efficient classifier requires a robust feature extraction method that generates stable image representations across various geometric transformations. Although Convolutional Neural Networks (CNNs) excel in spatial features, they often overlook rich frequency patterns in images. In this study, we explored the integration of the Wavelet Scattering Network to overcome the limitations of CNN-based models in achieving translation invariance and deformation stability. Our investigation focuses on assessing the efficacy of this approach by incorporating it into the EfficientFormerV2 architecture [1] in three different model variants. Our proposed method achieves a 1.68 % improvement in top-1 accuracy compared to EfficientFormerV2 on ImageNet100, a subset of ImageNet-1k comprising 1k randomly selected classes. We believe that integrating the properties of wavelet scattering transforms into existing models can yield more efficient and robust feature representations.

Index Terms— Feature extraction, Translation invariance, Deformation stability, Frequency domain, Wavelet Transform, Scattering network, EfficientFormerV2

1. INTRODUCTION

Efficient feature extraction methods are critical in pattern recognition of various visual understanding tasks. Over the last decade, CNNs have dominated the field of computer vision because of their capacity to learn spatially hierarchical feature representations of images. Despite the success of extracting spatial features, images may also inherently contain rich frequency patterns useful for visual recognition tasks. Recent works have explored the potential of learning in the frequency domain, employing techniques such as the Fourier Transform and Wavelet Transform [2, 3, 4, 5, 6, 7, 8, 9, 10, 11]. This transformation can enhance the capability of feature representations, thereby improving the overall performance of a model.

Images contain vital characteristics distributed across different frequency bands that can be easily captured in the frequency domain. These include patterns, low-frequency components (global structures), and high-frequency components (fine details). Incorporating frequency information with spatial features helps preserve global context while capturing local details, enhancing the representation learning capabilities. Additionally, frequency domain representations are often more robust to certain transformations, such as changes in scale and rotation, which improve the generalization of a model. Therefore, extracting features from the frequency domain can be more efficient in learning the underlying structure of the data.

Due to convolution and/or pooling operations being built into CNNs, it is widely thought that CNNs are inherently translation-invariant. However, several studies have shown that these networks consistently fail to recognize objects with a new location that diverges from the training data [12]. While augmenting the data set with fully translated samples can alleviate the limitation to some extent, it does not address the underlying architectural invariance issue. In addition, CNNs may struggle to capture complex deformations, especially in scenarios involving significant variations in scale, rotation, or non-rigid transformation. Despite the effectiveness of convolutional layers and pooling operations in enhancing the deformation stability of CNNs, the fixed kernel size of convolutional filters may not adequately adapt to such variations as they are designed to capture simple spatial relationships.

In contrast, scattering transform [13] is a more efficient alternative technique to CNNs that ensures both stability to deformations and translation invariance. The scattering network utilizes wavelet transform filters to decompose the input image into frequency bands. It then applies modulus nonlinearities and averaging operations to extract translation-invariant features that are stable to deformations. By integrating the scattering network within existing CNN models at various stages, such as initial, middle, or deeper layers, the model can benefit from both the properties of the scattering transform—such as stability to deformations and translation invariance—and the hierarchical representations learned by the convolutional layers. This combination of learning in the spatial and scattering transform spaces improves the representation learning and generalization capabilities.

Contributions: Our key contributions can be summarized as follows:

- We propose the integration of the wavelet scattering transform to improve the feature representation to be translation invariant and stable to deformation.
- We demonstrate superior performance by applying the wavelet scattering transform to EfficientFormerV2, surpassing the original design on ImageNet100.

2. RELATED WORK

Wavelet transform is a mathematical tool that decomposes signals or images into different frequency components, allowing for a localized analysis in both time and frequency domains. Several studies have explored the application of wavelet transform in computer vision tasks. In a study by [6], a multi-resolution analysis is combined with Haar wavelet CNNs for texture classification and image annotation tasks. Multi-level wavelet CNN (MWCNN) [4] introduces multi-level wavelet transform to enlarge the receptive field without information loss in image restoration. In addition, Bae et al. [2] demonstrates the effectiveness of learning CNN representations over wavelet sub-bands for image restoration tasks.

Moreover, Bruna & Mallat [13] propose a hybrid architecture that integrates a scattering network within the initial layers of ResNet, achieving competitive performance with fewer parameters. In the context of Transformer-based models, Scattering Vision Transformer (SViT) [14] explores the utilization of scattering transform for tokenization to enhance feature extraction capabilities in Vision Transformer (ViT) models. DWTFormer [10] introduces a novel framework incorporating Discrete Wavelet Transform into the Transformer architecture, improving feature representations and achieving superior performance in visual recognition tasks compared to other Transformer-based models. Yao et al. [15] implements invertible down-sampling using wavelet transforms over the keys and values to address the quadratic cost of self-attention. More recently, the Scattering Vision Transformer (SVT) [16] utilizes a scattering network based on the Dual-Tree Complex Wavelet Transform (DTCWT) to address attention mechanism complexity. SVT achieves state-of-the-art performance on various computer vision tasks while reducing parameters and computational complexity.

3. PRELIMINARIES

In this section, we revisit the scattering network [13]. The scattering network is widely used for signal analysis and has been found useful in image processing. It is based on wavelet transforms, which decompose a signal or an image into different frequency sub-bands. This decomposition helps capture both high and low-frequency information of the input data.

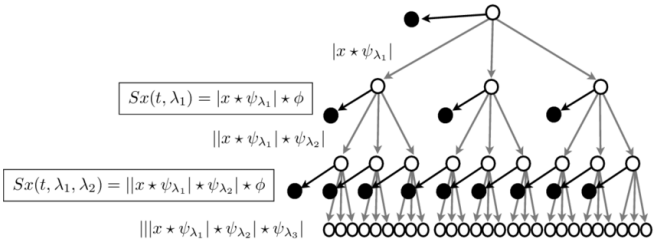

Fig. 1. A cascaded scattering transform architecture is computed by iterating on wavelet transform and modulus operators. The output of the modulus is averaged by ϕ to get the scattering coefficients (black arrows)

The scattering network performs multiple wavelet transform convolutions with non-linear modulus and averaging operators in a cascaded manner.

The scattering network extracts translation-invariant descriptors at multiple layers from the input image x by utilizing the wavelet transform. By scaling and rotating a single band-pass filter ψ, two-dimensional directional wavelets are obtained. For any $j \in \mathbb{Z}$ and rotation $r \in G$, where G is a discrete and finite set of rotations, multiscale directional wavelet filters are defined by

$$\psi_\lambda(u) = 2^j \psi(2^j r^{-1} t), \tag{1}$$

where $\lambda = (2^j, r)$ and $\psi(u) = \psi^a(u) + i\psi^b(u), u = (u_1, u_2)$.

The input image x is convolved by a filter bank of dilated and rotated wavelets $\psi_\lambda(u)$ formulated as $\{x * \psi_\lambda(u)\}_\lambda$

A wavelet transform is not inherently translation-invariant since it commutes with translations. To introduce a translation invariant representation, the modulus nonlinearity is first applied, followed by an averaging operation:

$$|x * \psi_{\lambda_1}(u)| = \sqrt{(x * \psi^a_{\lambda_1}(u))^2 + (x * \psi^b_{\lambda_1}(u))^2} \tag{2}$$

The modulus acts as a pooling operation that compresses the signal's energy. Next, the output result of the modulus operator is averaged by $\phi(u)$, a low-pass filter, to obtain the first-order scattering coefficients that are translation invariant representations, denoted as $S_1 x(u)$.

$$S_1 x(u) = |x * \psi_{\lambda_1}(u)| * \phi(u), \tag{3}$$

Higher-order coefficients can be computed by iteratively applying wavelet transforms and modulus operators. Wavelet coefficients are calculated for frequencies $2^j \geq 2^{-J}$ at a maximum scale of 2^J. Frequencies lower than this are filtered using $\phi_{2^J}(u) = 2^{-J/2}\phi(2^{-J}u)$.

The following process can derive the scattering coefficients S_{Jx} for the network at various scales and orientations for several layers.

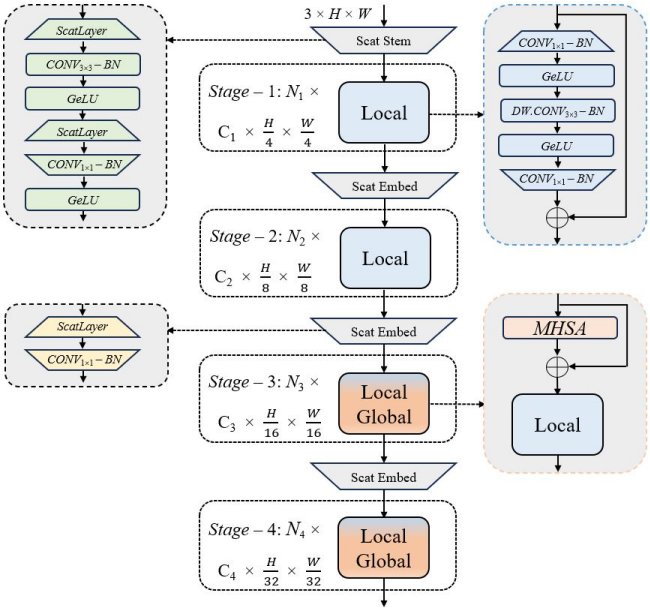

Fig. 2. Overview of the EfficientFormerV2 network with scattering network integrated into the stem (Scat Stem) and embedding layers (Scat Embed). For more details about the network, refer to [1]

$$S_J x = \begin{pmatrix} x * \phi_{2^J} \\ |x * \psi_{\lambda_1}| * \phi_{2^J} \\ ||x * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \phi_{2^J} \\ |||x * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \psi_{\lambda_3}| * \phi_{2^J} \\ \dots \end{pmatrix} \lambda_1, \lambda_2, \lambda_3, \dots$$

4. METHODOLOGY

This section provides an in-depth integration design of the scattering network into the EfficientFormerV2 architecture [1], explicitly optimizing the Stem and Embedding layers.

4.1. Scattering Stem Layer (Scat Stem)

The initial layer of a network, commonly known as the stem, typically serves as the first stage in feature extraction. Its primary purpose is to extract low-level features by significantly reducing the resolution of the input image to achieve an appropriate size for the feature map. In the context of the EfficientFormerV2 model, the stem comprises two stride convolutional layers that downsample the image, resulting in a resolution of $\frac{H}{4} \times \frac{W}{4}$. Although this method can achieve the desired feature map size, it can lead to losing spatial information and fine-grained details.

To mitigate these limitations, we propose integrating the scattering network within the stem layer. Incorporating the scattering network in the stem layer provides a lossless downsampling alternative while leveraging better low-level feature representations. This integration allows us to extract rich and meaningful features while maintaining the desired resolution for subsequent stages of the architecture. Scat stem can be formulated as:

$$X_0' = \text{Scat}((X_0)) \tag{4}$$

$$X_0'' = \text{Scat}(\sigma(\text{Conv-BN}_{3\times3}^{eC\to eC}(X_0')) \tag{5}$$

$$X_1 = \sigma(\text{Conv-BN}_{1\times1}^{e^2C\to C_1}(X_0'')), \tag{6}$$

where e is seven, which refers to the six high-pass sub-bands, in addition to the low-pass outputs of the scattering transform (Scat). The intended embedding dimension is given by C_1. The BN refers to BatchNorm, and σ represents an activation function. The final output result is denoted as $X_1 \in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times C_1}$.

4.2. Scattering Embedding Layer

The embedding layer in a network is typically responsible for transforming the feature space into higher dimensions while simultaneously reducing the input resolution. This reduction in spatial dimensions is crucial to balance the increased number of parameters resulting from the higher feature dimension. Instead of using a stride convolution to reduce spatial dimensions and expand channel numbers, we propose the integration of the scattering network to achieve a localized resolution reduction and channel expansion. The scattering embedding layer (Scat Embed) can be summarized as:

$$Y_i = \text{Scat}(\text{Conv-BN}_{1\times1}(X_{i|i>1})) \tag{7}$$

4.3. Model Design Variations

We investigate the effectiveness of our method with different model variations.

Variant 1: We added a 3×3 Conv layer before the scattering transform in the Scat Stem layer along with a reduction ratio variable denoted as r to control the intended embedding dimension. The initial layer can be formulated as follows:

$$X_0' = \text{Scat}(\sigma(\text{Conv-BN}_{3\times3}^{3\to\frac{C_1}{r}}(X_0)), \tag{8}$$

we refer to this modified model as ScatFormer-1-S0.

Variant 2: We started with a high embedding dimension and reduced it through the Scat Embed. Therefore, we swapped the ScatLayer and the 1×1 Conv layer. First, the input to the 1×1 Conv will reduce the input channels by a reduction variable, which will be expanded by 7 through the ScatLayer. We removed the 1×1 Conv layers in the Local Block, keeping only the DW CONV-BN and the GeLU layers. We will

Table 1. **Classification results on ImageNet-100 [17]**. Number of parameters (Params M), GMACs, Training Epochs, and Top-1 accuracy are reported for the two models

Model	Params (M)	GMACs	Top-1 (%)
EfficientFormerV2-S0 [1]	3.26	0.39	86.78
ScatFormer-S0 (Ours)	3.22	0.43	88.46
ScatFormer-1-S0 (Ours)	3.25	0.77	88.00
ScatFormer-2-S0 (Ours)	4.90	0.48	84.84

see that dropping all the 1×1 Conv layers impaired the performance since the 1×1 Conv layer after the DW CONV is necessary to learn cross-channel interactions. This method is denoted as ScatFormer-2-S0.

5. EXPERIMENTS

5.1. Classification Setup

We modified the EfficientFormerV2 [1] code repository to integrate the scattering transform. The dataset represents a subset of ImageNet-1K [18], consisting of randomly selected 100 classes. The training set has 1300 images for each class, while the testing set includes 50 images for each class. The training setup closely follows that of EfficientFormerV2 [1]. The original model and the modified version were trained from scratch for 200 epochs using an 8xRTX4090 on Imagenet-100 [17] using the AdamW optimizer. The learning rate is set to 10^{-3} with a batch size of 256 and cosine decay. The image resolution is fixed at 224×224 for training and testing. Noteworthy, we opted not to use the distillation technique proposed in DeiT [19], used in EfficientFormerV2 [1].

5.2. ScatFormer Evaluation

Table 1 compares our proposed model modification and the original one. Due to limited resources, we picked only the smaller versions of the models for comparison. ScatFormer-S0 outperforms EfficientFormerV2-S0 by 1.68%, with 0.04M fewer parameters. Although integrating scattering transform can add slightly more complexity (GMACs), it increases the accuracy and reduces the model's size. For ScatFormer-1-S0, adding 3×3 Conv layer in the Stem Layer would not improve the performance compared to ScatFormer-S0 but rather almost double the complexity. Lastly, ScatFormer-2-S0 does not do well since removing the point-wise Conv after the depth-wise Conv prevents the network from capturing information from different channels.

6. CONCLUSION

This work investigated incorporating the Scattering Network into EfficientFormerV2 architecture for improved feature extraction and stability. By integrating the scattering network at various stages of the architecture, such as initial, middle, or deeper layers, we demonstrate a 1.68% improvement in top-1 accuracy on ImageNet100, surpassing the performance of the original design. The results showcased the potential of frequency domain representations in preserving global context and capturing local details, leading to improved representation learning and generalization capabilities.

The primary object of this paper is to show the potential of our proposed method. Therefore, we have limited the scope of the experiments to a small model and a subset of ImageNet due to limited resources. In future work, we plan to assess our approach using the larger model size of EfficientFormerV2 and train it on ImageNet-1K.

7. REFERENCES

[1] Yanyu Li, Ju Hu, Yang Wen, Georgios Evangelidis, Kamyar Salahi, Yanzhi Wang, Sergey Tulyakov, and Jian Ren, "Rethinking vision transformers for mobilenet size and speed," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 16889–16900.

[2] Woong Bae, Jaejun Yoo, and Jong Chul Ye, "Beyond deep residual learning for image restoration: Persistent homology-guided manifold simplification," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 145–153.

[3] Tiantong Guo, Hojjat Seyed Mousavi, Tiep Huu Vu, and Vishal Monga, "Deep wavelet prediction for image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 104–113.

[4] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo, "Multi-level wavelet-cnn for image restoration," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 773–782.

[5] Travis Williams and Robert Li, "Wavelet pooling for convolutional neural networks," in *International conference on learning representations*, 2018.

[6] Shin Fujieda, Kohei Takayama, and Toshiya Hachisuka, "Wavelet convolutional neural networks," *arXiv preprint arXiv:1805.08620*, 2018.

[7] Edouard Oyallon, Eugene Belilovsky, and Sergey Zagoruyko, "Scaling the scattering transform: Deep hybrid networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5618–5627.

[8] Fergal Cotter and Nick Kingsbury, "A learnable scattternet: Locally invariant convolutional layers," in *2019 IEEE international conference on image processing (ICIP)*. IEEE, 2019, pp. 350–354.

[9] Lu Chi, Borui Jiang, and Yadong Mu, "Fast fourier convolution," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4479–4488, 2020.

[10] Dongwook Yang and Seung-Woo Seo, "Discrete wavelet transform meets transformer: Unleashing the full potential of the transformer for visual recognition," *IEEE Access*, 2023.

[11] Badri N Patro, Vinay P Namboodiri, and Vijay Srinivas Agneeswaran, "Spectformer: Frequency and attention is what you need in a vision transformer," *arXiv preprint arXiv:2304.06446*, 2023.

[12] Avraham Ruderman, Neil C Rabinowitz, Ari S Morcos, and Daniel Zoran, "Pooling is neither necessary nor sufficient for appropriate deformation stability in cnns," *arXiv preprint arXiv:1804.04438*, 2018.

[13] Joan Bruna and Stéphane Mallat, "Invariant scattering convolution networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1872–1886, 2013.

[14] Tianming Qiu, Ming Gui, Cheng Yan, Ziqing Zhao, and Hao Shen, "Svit: Hybrid vision transformer models with scattering transform," in *2022 IEEE 32nd International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2022, pp. 01–06.

[15] Ting Yao, Yingwei Pan, Yehao Li, Chong-Wah Ngo, and Tao Mei, "Wave-vit: Unifying wavelet and transformers for visual representation learning," in *European Conference on Computer Vision*. Springer, 2022, pp. 328–345.

[16] Badri N Patro and Vijay Srinivas Agneeswaran, "Scattering vision transformer: Spectral mixing matters," *arXiv preprint arXiv:2311.01310*, 2023.

[17] Ambesh Shekhar, "Imagenet100," Aug 2021.

[18] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al., "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, pp. 211–252, 2015.

[19] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou, "Training data-efficient image transformers & distillation through attention," in *International conference on machine learning*. PMLR, 2021, pp. 10347–10357.