Fraesormer: Learning Adaptive Sparse Transformer for Efficient Food Recognition

Shun Zou[1,7], Yi Zou[2], Mingya Zhang[3], Shipeng Luo[4], Zhihao Chen[5], Guangwei Gao[6,7†]
[1]Nanjing Agricultural University [2]Xiangtan University [3]Nanjing University [4]Northeast Forestry University
[5]Beijing Information Science and Technology University [6]Nanjing University of Posts and Telecommunications
[7]Soochow University

2025 IEEE International Conference on Multimedia and Expo (ICME) | 979-8-3315-9495-4/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICME59968.2025.11209119

***Abstract*—In recent years, Transformer has witnessed significant progress in food recognition. However, most existing approaches still face two critical challenges in lightweight food recognition: (1) the quadratic complexity and redundant feature representation from interactions with irrelevant tokens; (2) static feature recognition and single-scale representation, which overlook the unstructured, non-fixed nature of food images and the need for multi-scale features. To address these, we propose an adaptive and efficient sparse Transformer architecture (Fraesormer) with two core designs: Adaptive Top-k Sparse Partial Attention (ATK-SPA) and Hierarchical Scale-Sensitive Feature Gating Network (HSSFGN). ATK-SPA uses a learnable Gated Dynamic Top-K Operator (GDTKO) to retain critical attention scores, filtering low query-key matches that hinder feature aggregation. It also introduces a partial channel mechanism to reduce redundancy and promote expert information flow, enabling local-global collaborative modeling. HSSFGN employs gating mechanism to achieve multi-scale feature representation, enhancing contextual semantic information. Extensive experiments show that Fraesormer outperforms state-of-the-art methods. code is available at https://zs1314.github.io/Fraesormer.**

***Index Terms*—Efficient Food Recognition, Sparse Transformer, Selective Dynamic Attention, Gated Dynamic Top-k Operator**

I. Introduction

Food plays a central role in our daily lives, and food computing can assist in areas such as diet, health, and the food industry [1], [2]. With the rise of deep learning, food computing has gained increasing attention and undergone significant development in computer vision and multimedia [3]. Furthermore, as the primary goal of food computing systems is to help individuals manage their diet and health while enhancing their daily activities, it is essential to develop an efficient system specifically designed for food image recognition on edge devices [4].

Many learning-based methods have recently utilized various CNN architectures for food recognition [5], [6]. Although CNNs achieve good efficiency and generalization due to their local connections and translation invariance, the receptive field of the convolution operator is limited, preventing them from capturing global dependencies and modeling long-range pixel relationships. In food recognition, food images often contain multiple ingredients that are sparsely distributed, adding complexity and requiring fine-grained analysis [11]. Therefore, there is a greater demand for neural networks capable of extracting long-range relationships [4]. To address this challenge, the self-attention mechanism offers a more powerful alternative [12]. Self-attention is the core component of Transformers, showing state-of-the-art performance in computer vision [13], [14]. Researchers have also applied Transformers to food recognition [15], achieving promising results due to their ability to extract non-local information and model global states effectively. However, while Transformers excel at capturing long-range pixel interactions, their complexity increases quadratically with spatial resolution, resulting in significant computational burdens. This poses a considerable challenge for developing a lightweight recognition system. To date, research on lightweight Transformer-based models specifically tailored for food recognition remains limited.

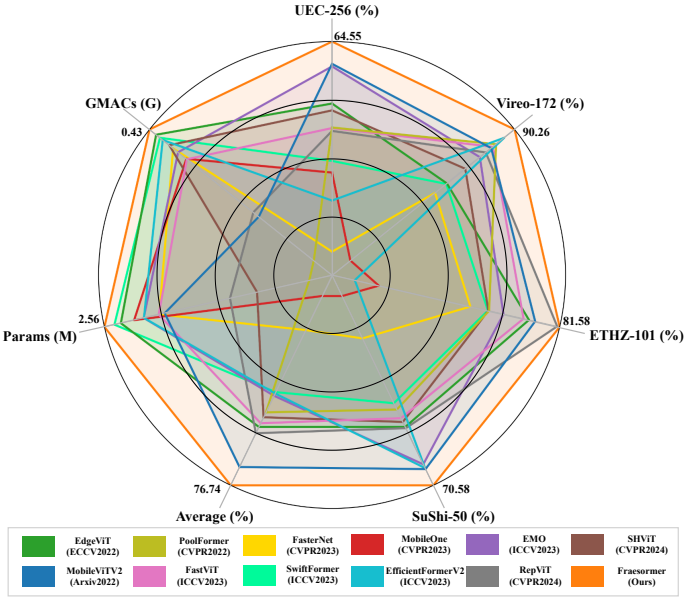

Fig. 1: A seven-dimensional radar map of the Top-1 Acc of ETHZ-101 [7], Vireo-172 [8], UEC-256 [9], SuShi-50 [10], along with Average Acc, Params, and GMACs.

To address these issues, we re-examine the challenges associated with lightweight food recognition: (1) Although some studies have attempted to simplify the attention mechanism

†Corresponding author. Email: zs@stu.njau.edu.cn, csgwgao@njupt.edu.cn. This work was supported in part by the Open Fund Project of Provincial Key Laboratory for Computer Information Processing Technology (Soochow University) under Grant KJS2274.

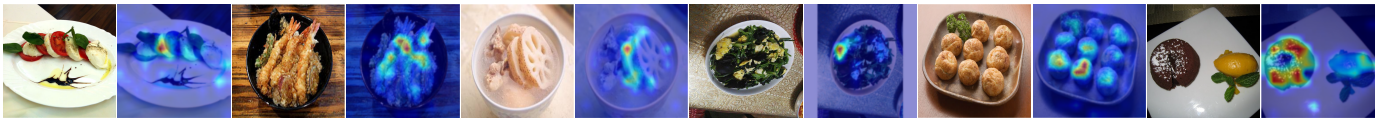

Fig. 2: Visualization of the impact of each spatial location on the final prediction of the DeiT-S model [16], [17]. The results show that the final prediction of the vision transformer is primarily based on the most influential tokens, indicating that a large portion of tokens can be removed without affecting performance.

[14], [18], the number of parameters and computational costs remain high. Moreover, we observe that standard Transformers aggregate features based on all attention relationships between queries and keys; however, the tokens in the keys are not always relevant to those in the queries [19] (see Fig. 2). Using irrelevant tokens to estimate self-attention values leads to computational overhead, information redundancy, and noisy interactions during feature aggregation; (2) Traditional visual recognition methods struggle to handle the unstructured and variable nature of food images, necessitating an adaptive feature extraction paradigm [20]. For example, the appearance of food can change significantly during cooking; the visual features of raw fish slices differ greatly from those of cooked fish fillets. Traditional feature extraction methods cannot easily adapt to such dynamic changes. Moreover, some dishes can be easily confused due to similarities in cooking methods. For instance, "beef curry" and "braised beef with potatoes" may be misclassified because of similar preparation techniques ("curry braise" and "potato braise"). Therefore, an adaptive feature extraction mechanism is more effective in addressing challenges such as changes in ingredient morphology, regional differences, and similar cooking styles; (3) Multi-scale features are crucial for accurately recognizing food semantics. In a French dish, both the main ingredient and small components contribute to its overall characteristics. For example, in a foie gras dish, while the large foie gras portion is the main element, the small fruits and herbs are also key to identification. If the model focuses solely on the main ingredient, it may fail to capture complete semantic information, leading to misinterpretation.

To address the above challenges, we propose a **f**ood **r**ecognition-focused **a**daptive **e**fficient **s**parse Trans**former** network, named **Fraesormer**. Specifically, the key component of this framework is the Fraesormer Block, which includes Adaptive Top-k Sparse Partial Attention (ATK-SPA) for extracting the most significant attention values and the Hierarchical Scale-Sensitive Feature Gating Network (HSSFGN) for exploring multi-scale contextual local information. More precisely, ATK-SPA does not compute dense attention matrices using all tokens. Instead, it adapts the value of k based on the input features by using a novel and efficient Gated Dynamic Top-K Operator (GDTKO). It selects the top-k tokens with the highest attention values to filter out irrelevant noise, fully leveraging the network's sparsity while ensuring a dynamic feature extraction paradigm. Additionally, we have combined a parallel partial channel mechanism with the attention mechanism, effectively addressing channel redundancy while enhancing local information. This approach utilizes two sets of complementary features to collaboratively model both local and non-local features. Furthermore, the HSSFGN we designed further exploits cross-level multi-scale information for improved feature aggregation. Finally, comprehensive experiments show that Fraesormer achieves the better performance-parameter trade-off, making it a true "Heptagon Warrior", as shown in Fig. 1. The main contributions of this work are summarized as follows:

- We propose an efficient adaptive sparse Transformer architecture that promotes the flow of the most valuable information across layers, effectively addressing the challenges faced in lightweight food recognition.
- We introduce Adaptive Top-k Sparse Partial Attention, which adaptively reduces redundant information and alleviates computational burden. Additionally, we design a hybrid-scale feedforward network based on the gating mechanism to efficiently explore cross-level multi-scale representations.
- We conducted comprehensive experiments on four widely used authoritative food datasets, and the results demonstrate that our method outperforms state-of-the-art approaches. Furthermore, we provide extensive ablation studies to highlight the contributions of the design.

II. Method

A. Overall Pipeline

The overall pipeline of our Fraesormer is illustrated in Fig. 3. In line with previous general visual backbones [21], the Fraesormer is divided into four stages, each consisting of several Fraesormer Blocks. Additionally, each stage is preceded by a merging layer for spatial downsampling and channel expansion. We apply a global average pooling layer to the final output and send it to a linear classification head. Similar to earlier backbone networks [22], [23], we incorporate CPE [24] into our model. In each block, we have developed ATK-SPA, which differs from standard self-attention in Transformers, to achieve feature sparsity and execute the feature aggregation process more efficiently. We also introduce a parallel partial channel mechanism to reduce redundant information and enhance the retention of local features, facilitating the expansion of channel information. Furthermore, we integrate HSSFGN within the Fraesormer Block to extract rich local scale information across different levels while filtering out hidden redundant information in the feature maps. Formally, given the input X_{l-1} of the $l-1$ block, the processing of the Fraesormer Block is defined as follows:

$$X_l = X_{l-1} + DW(X_{l-1}), \tag{1}$$

$$X_l' = X_l + ATK\text{-}SPA(LN(X_l)), \tag{2}$$

$$X_l'' = X_l' + HSSFGN(LN(X_l')), \tag{3}$$

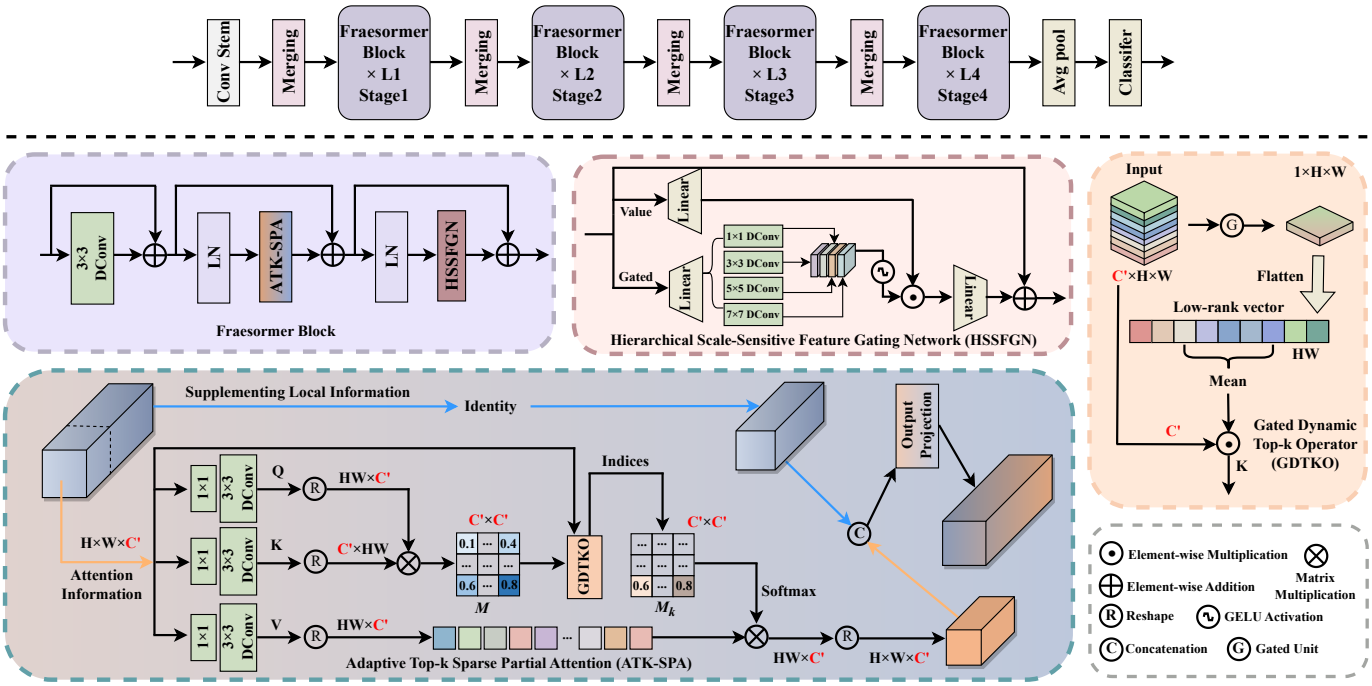

Fig. 3: The overall architecture of the proposed Fraesormer.

where $DW(\cdot)$ represents the depth-wise convolution operation, $LN(\cdot)$ denotes layer normalization, and X_l, X_l', X_l'' are the outputs of DW, ATK-SPA, and HSSFGN, respectively.

B. Adaptive Top-k Sparse Partial Attention

Our ATK-SPA is motivated by two main objectives: (1) to extract only the most critical information during attention calculation, thereby eliminating the influence of noise tokens while minimizing redundant information to reduce computational burden; and (2) to achieve adaptive aggregation and dynamic filtering of image features, proposing a feature extraction strategy that flexibly adapts to the complexity of food images and the irregularity of visual patterns, effectively addressing diverse appearances and semantic content.

Fig. 3 illustrates the architecture of ATK-SPA. Specifically, given the input normalized feature $F \in \mathbb{R}^{H\times W\times C}$, we selectively choose a subset of channels for spatial feature aggregation while keeping the remaining channels unchanged, resulting in $X_{att} \in \mathbb{R}^{H\times W\times C'}$ and $X_{sup} \in \mathbb{R}^{H\times W\times C''}$. Next, we apply a 1×1 point-wise convolution on X_{att} to aggregate local information across pixel channels, and use 3×3 depth-wise convolutions to enhance spatial context across channels, generating the matrices for query Q, key K, and value V. We then perform a dot-product operation on the reshaped Q and K to produce a dense attention matrix $M \in \mathbb{R}^{C'\times C'}$. In contrast to calculating a spatial matrix with dimensions $HW\times HW$, measuring channel similarity helps reduce memory consumption, thereby enabling efficient inference [25]. We re-examine the standard dense self-attention mechanism (SDSA) [12] used in most previous work, which computes the attention map for all query-key pairs:

$$SDSA = Softmax(\frac{QK^T}{\sqrt{d}} + B)V. \tag{4}$$

Here, $\sqrt{d}$ represents an optional temperature coefficient used to control the size of the dot product between Q and K before applying softmax function, and B denotes a learnable relative position bias. However, not all tokens in the queries are closely related to their corresponding tokens in the keys, so calculating and utilizing the similarities of all query-key pairs can lead to noisy interactions and information redundancy. To address this issue, we perform adaptive selection on the dense attention matrix M, retaining only the top k attention scores with the highest contributions. This approach aims to preserve the most useful information in the attention matrix while filtering out the majority of noise interference. The selection of k is critical; instead of fixing k, we adaptively adjust it based on the input using a novel, simple yet efficient Gated Dynamic Top-K Operator. This enables a dynamic selection process that transitions the attention matrix from dense to sparse (which will be elaborated upon in the next subsection). For example, with $k = 2/3$, we retain only the top $2/3$ of the highest attention scores while masking the remaining elements to zero. Specifically, we generate a binary mask matrix:

$$[M_k]_{ij} = \begin{cases} 1 & M_{ij} \in indices_k \\ 0 & \text{otherwise} \end{cases}, \tag{5}$$

where indices_k represents the indices of the top k highest values. Formally, ATK-SPA can be described as:

$$X_{att}, X_{sup} = Split(X), \tag{6}$$

$$ATK\text{-}SPA = Softmax(T_k(\frac{QK^T}{\sqrt{d}} + B))V, \tag{7}$$

$$\hat{X_{att}} = ATK\text{-}SPA(X_{att}W_d^Q W_p^Q, X_{att}W_d^K W_p^K, X_{att}W_d^V W_p^V), \tag{8}$$

$$X' = Concat(\hat{X_{att}}, X_{sup})W_p^O, \tag{9}$$

where $Split(\cdot)$ denotes channel-wise splitting, $W(\cdot)_p$ is the 1×1 point-wise convolution, $W(\cdot)_d$ is the 3×3 depth-wise

TABLE I: Comparison of quantitative results on four food datasets. **Bold** values indicate the best result. ↑ means higher is better, ↓ means lower is better.

Model	Params (M)↓	MACs (G)↓	UEC-256 [9] Top-1(%)↑	Vireo-172 [8] Top-1(%)↑	ETHZ-101 [7] Top-1(%)↑	SuShi-50 [10] Top-1(%)↑	Average Top-1(%)↑	Year
EMO-1M [26]	1.36	0.26	62.882	87.560	80.267	54.096	71.201	ICCV2023
RepViT-M0-6 [27]	2.49	0.40	62.018	87.693	79.228	55.060	71.000	CVPR2024
MobileNetV3-Small [28]	2.94	0.07	54.351	84.343	74.426	51.807	66.232	ICCV2019
SwiftFormer-XS [29]	3.48	0.62	58.099	87.789	79.649	62.651	72.047	ICCV2023
EfficientFormerv2-S0 [13]	3.60	0.40	53.055	88.323	70.203	68.072	69.913	ICCV2023
FasterNet-T0 [30]	3.91	0.34	52.021	85.583	76.386	53.976	66.992	CVPR2023
FastViT-T8 [14]	4.03	0.56	58.161	88.659	79.040	59.398	71.312	ICCV2023
EdgeViT-XXS [31]	4.07	0.55	61.200	87.811	81.252	64.940	73.801	ECCV2022
Mobileone-S0 [32]	5.29	1.10	57.467	84.321	74.733	52.294	67.204	CVPR2023
MobileNetV3-Large [28]	5.48	0.24	61.077	87.789	79.891	61.807	72.641	ICCV2019
Fraesormer-Tiny (Ours)	2.56	0.43	**64.548**	**90.257**	**81.584**	**70.578**	**76.742**	–
SwiftFormer-S [29]	6.09	1.01	59.488	88.851	79.767	66.988	73.774	ICCV2023
GhostNetV2-1.0 [33]	6.16	0.18	58.917	87.981	80.144	56.265	70.827	NIPS2022
EMO-6M [26]	6.17	0.95	63.206	89.002	80.144	68.554	72.227	ICCV2023
EfficientFormerV2-s1 [13]	6.19	0.67	55.955	89.877	73.693	68.916	72.110	ICCV2023
SHViT-S1 [34]	6.33	0.25	57.621	86.224	76.450	60.723	70.255	CVPR2024
EdgeViT-XS [31]	6.77	1.13	62.465	89.810	82.762	70.120	76.289	ECCV2022
FastViT-T12 [14]	7.55	1.11	59.873	89.471	81.015	64.096	73.614	ICCV2023
FasterNet-T1 [30]	7.60	0.86	53.193	87.383	78.708	56.386	68.918	CVPR2023
EfficientNet-B1 [35]	7.79	0.60	65.659	90.386	82.480	69.036	76.890	ICML2019
MobileOne-S2 [32]	7.88	1.35	63.067	88.268	80.950	61.928	73.553	CVPR2023
MobileViTV2-1.3 [18]	8.10	2.42	63.345	89.412	81.520	69.012	75.822	Arxiv2022
GhostNetV2-1.3 [33]	8.96	0.29	60.753	89.028	81.163	59.639	72.646	NIPS2022
Fraesormer-Base (Ours)	6.39	1.21	**68.050**	**90.799**	**83.054**	**76.024**	**79.482**	–
SHViT-S2 [34]	11.48	0.37	57.282	87.176	76.584	61.259	70.575	CVPR2024
FastViT-Sa12 [14]	11.58	1.52	62.789	89.523	81.322	64.819	74.613	ICCV2023
PoolFormer-S12 [36]	11.92	1.83	52.823	85.996	77.490	61.264	69.393	CVPR2022
EfficientFormerV2-S2 [13]	12.71	1.27	50.741	88.504	59.797	70.120	67.291	ICCV2023
EdgeViT-S [31]	13.11	1.91	61.031	88.401	82.624	68.313	75.092	ECCV2022
RepViT-M1-5 [27]	14.10	2.32	59.719	89.216	82.475	65.060	74.118	CVPR2024
SHViT-S4 [34]	16.59	0.79	60.830	88.475	79.490	64.458	73.313	CVPR2024
PoolFormer-S24 [36]	21.39	3.42	59.904	89.589	79.520	63.256	73.067	CVPR2022
FasterNet-S [30]	31.18	4.57	51.095	84.741	80.248	61.234	69.330	CVPR2023
Fraesormer-Large (Ours)	10.39	1.74	**67.957**	**91.515**	**85.460**	**75.542**	**80.119**	–

TABLE II: Ablation experiments of different components.

Model	Params (M)↓	MACs (G)↓	Top-1 (%)↑
w/o ADK-SPA	1.99	0.33	62.715
w/o HSSFGN	1.52	0.24	60.077
ours	2.56	0.43	64.548

convolution, and $W(\cdot)_p^O$ denotes the projection weights for the output layer. Concat$(\cdot)$ denotes the concatenation operation, while $T_k(\cdot)$ represents the Gated Dynamic Top-K operator. It is important to note that, similar to traditional multi-head self-attention [12], we divide the channels into "heads" and compute separate attention maps in parallel. Additionally, we apply the 1×1 point-wise convolution as the final projection layer across all channels, not just the initially divided attention channels. This ensures that the attention features extend to other channels while enhancing the local context, preserving information, and facilitating channel fusion and collaboration.

Gated Dynamic Top-k Operator (GDTKO). Since using a fixed k lacks the flexibility and dynamic adaptability needed to explore the potential of sparsity, we propose a Top-K operator that dynamically allocates and modulates based on the input. This allows the model to optimize attention distribution under both sparse and dense information conditions. The architecture of GDTKO is shown in Fig. 3. Given the normalized feature $F \in \mathbb{R}^{H \times W \times C'}$ from partial channels, we first use a Gated Unit to evaluate the contribution of the input feature map, capturing key information while suppressing noise. This produces a supervision modulation feature that retains the same spatial dimensions but contains only one channel. We then flatten this feature and compute its average value. Finally, we multiply the obtained supervised average by the channel dimension C' of the input feature F. This ensures a proportional relationship between the value of k and the feature dimension, preventing imbalanced performance in low- or high-dimensional cases. Moreover, this enables the model to automatically adjust its behavior as the feature dimension changes, thereby improving its generalization ability.

C. Hierarchical Scale-Sensitive Feature Gating Network

The standard feed-forward network (FFN) usually processes information at each pixel independently, which is crucial for the self-attention mechanism [12]. However, it is limited by single-scale feature extraction and overlooks the significance of multi-scale information in food recognition. Traditional multi-scale feed-forward networks tend to be bulky and redundant, requiring more computational resources. Therefore, we propose a more elegant approach to capture multi-scale local contextual information. Specifically, we design a scale-aware feed-forward network with a gate-like mechanism, incorporating depthwise separable convolutions at different scales (see Fig. 3). On one hand, the gating mechanism alleviates the burden of processing redundant information, aligning with the design philosophy of channel attention by focusing on important features in the channel dimension. On the other hand, the proposed HSSFGN controls the flow of information across layers, allowing each layer to concentrate on the fine details necessary for the other layers, thus achieving cross-layer expert information fusion. Formally, HSSFGN can be expressed as

$$X_G = X_V = Linear(X_{l-1}), \tag{10}$$

$$X_{Gi} = Split(X_G), X'_{Gi} = f_{dw}^{i \times i}(X_{Gi}), i = [1, 3, 5, 7], \tag{11}$$

$$\hat{X_G} = \delta(Concat(X'_{Gi})), \tag{12}$$

$$X_l = Linear(X_V \otimes \hat{X_G}) + X_{l-1}, \tag{13}$$

where $f_{dw}^{k \times k}(\cdot)$ represents the $k \times k$ depth-wise convolution, $\delta(\cdot)$ denotes the GELU activation function, $\otimes$ indicates element-wise multiplication, and Linear$(\cdot)$ represents the linear layer.

III. Experiments and Analysis

A. Experimental Settings

Datasets. To evaluate the proposed model, we conducted experiments on four widely used food datasets: ETHZ Food-101 [7], Vireo Food-172 [8], UEC Food-256 [9], and SuShi-50 [10]. Additionally, we divided the datasets into training, validation, and test sets in a 6:2:2 ratio.

Training Details. We trained on images with a resolution of 224×224, and all models were trained from scratch using the AdamW optimizer, without any pretraining or fine-tuning. The learning rate was set to 1×10^{-3}, with a batch size of 256, for 300 epochs. We applied a cosine learning rate scheduler with a linear warm-up for the first 5 epochs. Data augmentation methods, including mixup augmentation and random erasing, were applied following the techniques described in [34]. The entire framework was implemented in PyTorch and trained on an NVIDIA GeForce RTX 4090 GPU with 24GB of memory.

B. Comparisons with The State-of-the-arts

We compared our method with some open-source methods, including CNN-based approaches (RepViT [27], MobileNet v3 [28], FasterNet [30], MobileOne [32], EfficientNet [35], GhostNet v2 [33]), ViT-based methods (SwiftFormer [29], SHViT [34], PoolFormer [36]), and hybrid CNN-transformer approaches (EMO [26], EfficientFormer v2 [13], FastViT [14], EdgeViT [31], MobileViT v2 [18]). The experimental results are shown in Table I and Fig. 1. We categorize the

TABLE III: Ablation studies of different self-attention.

Model	Params (M)↓	MACs (G)↓	Top-1 (%)↑	Year
SDSA [12]	2.95	0.48	62.892	ICLR2021
L-MHSA [22]	2.95	0.56	63.524	CVPR2022
H-MHSA [37]	2.96	0.48	63.126	MIR2024
ATK-SPA	2.56	0.43	64.548	-

TABLE IV: Ablation studies of the PPCM.

Model	Params (M)↓	MACs (G)↓	Top-1 (%)↑
w/o PPCM	3.11	0.51	64.559
w/ PPCM	2.56	0.43	64.548
partial ratio = 1/8	2.21	0.39	62.921
partial ratio = 1/4 (ours)	2.56	0.43	64.548
partial ratio = 1/2	2.89	0.48	64.556

TABLE V: Ablation studies on alternatives to the HSSFGN.

Model	Params (M)↓	MACs (G)↓	Top-1 (%)↑	Year
SFFN [12]	2.97	0.48	62.805	ICLR2021
DFN [38]	5.44	0.59	64.982	CVPR2021
MixCFN [39]	3.05	0.51	63.144	CVPR2022
ConvGLU [40]	3.00	0.49	62.959	CVPR2024
HSSFGN	2.56	0.43	64.548	-

models into three groups based on parameter size. Within each parameter range, Fraesormer consistently outperforms other models in terms of Top-1 accuracy across all datasets while achieving a better balance between parameters and performance. Specifically, in the 0–$5M$ parameter range, Fraesormer-tiny achieves a Top-1 accuracy that is 6.83% higher than that of EfficientFormerV2-S0 [13], while saving 29% in parameters. It also surpasses MobileOne-S0 [32] by 9.54% in accuracy, while reducing parameters by 52% and computational cost by 62%. In the 6–$10M$ range, Fraesormer-Base's accuracy is 3.66% higher than that of MobileViTV2-1.3 [18], with a 21% reduction in parameters and a 50% decrease in computational cost. In the $> 10M$ parameter range, Fraesormer-Large achieves an accuracy that is 6.00% higher than that of RepViT-M1-5 [27], while saving 26% in parameters and 25% in computational cost. Additionally, it outperforms FasterNet-S [30] by 10.79% in accuracy, saving 67% in parameters and 62% in computational cost. Furthermore, unlike previous Transformer models [12], Fraesormer excels not only on large datasets (ETHZ Food-101 [7]) but also achieves optimal performance on smaller datasets (UEC-256 [9] and SuShi-50 [10]), indicating that Fraesormer can efficiently perform food recognition without requiring large data volumes or high computational costs.

C. Ablation Studies

Ablation Experiments with Different Components. To validate the effectiveness of various components in Fraesormer, we conducted ablation experiments on the UEC-256 dataset [9] (Unless otherwise specified, all subsequent experiments use this dataset). The results are shown in Table II. When we removed ATK-SPA, the model's accuracy decreased by 1.833%. Similarly, removing HSSFGN led to a 4.471% drop in accuracy. In Fraesormer, ATK-SPA adaptively extracts the most significant information in global contexts but cannot capture local contextual information. In contrast, HSSFGN focuses on capturing fine-grained local features across multiple scales and levels. Furthermore, HSSFGN and ATK-SPA complement each other, with HSSFGN suppressing redundancy in the channel dimension and ATK-SPA addressing redundancy in the spatial dimension. The collaboration of these two components enables Fraesormer to dynamically extract the most valuable insights at various levels while minimizing computational burden.

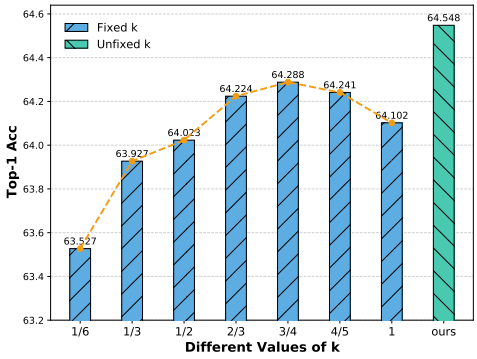

Fig. 4: Ablation analysis of different values of k.

TABLE VI: Ablation studies of the HSSFGN components.

Gating Mechanism	Multi-scale Conv	Params (M)↓	MACs (G)↓	Top-1 (%)↑
✗	✗	2.52	0.41	63.406
✓	✗	2.52	0.41	64.231
✗	✓	2.56	0.43	64.143
✓	✓	2.56	0.43	64.548

Adaptive Top-k Sparse Partial Attention. We replaced ATK-SPA with several recent efficient attention mechanisms: Standard Dense Self-Attention (SDSA) [12], Lightweight Multi-Head Self-Attention (L-MHSA) [22], and Hierarchical Multi-Head Self-Attention (H-MHSA) [37]. The quantitative results are shown in Table III. Compared to SDSA, L-MHSA, and H-MHSA, our ATK-SPA demonstrated superior performance in both accuracy and parameter efficiency.

Gated Dynamic Top-k Operator. The key factor for GDTKO is the value of k. To verify the effectiveness of the adaptive k value, we fixed different k values and compared them with the dynamic k value from GDTKO, as shown in Fig. 4. We observed that setting k to a smaller value led to a significant loss of useful information. Conversely, selecting a k value that is too large, or even including all tokens for attention computation, not only increased the computational burden but also introduced interference from irrelevant tokens. Thus, manually setting the k value fails to adaptively adjust the network based on input data, while GDTKO effectively balances sparsity and density to dynamically capture the most influential tokens.

Parallel Partial Channel Mechanism. To investigate the effectiveness of the partial channel mechanism, we removed this mechanism, and the experimental results are shown in Table IV. When the partial channel mechanism is applied, we achieve comparable accuracy with lower parameters and computational overhead. This demonstrates that the method enables token interactions across channels at a reduced cost, achieving a high cost-performance ratio. Furthermore, we explored the optimal ratio configuration. When the partial ratio is set to 1/4 (the default setting of Fraesormer), it achieves the best trade-off between accuracy and speed. Compared to very small values, a moderate increase in token interactions

across channels effectively improves performance at a low cost. However, excessively large values fail to provide a cost-effective performance improvement, as the accompanying computational cost outweighs the benefits.

Hierarchical Scale-Sensitive Feature Gating Network. To demonstrate the effectiveness of HSSFGN, we compared it with four variants: Standard Feed-Forward Network (SFFN) [12], Depth-wise Convolution Equipped Feed-forward Network (DFN) [38], Mixed-scale Convolutional Feedforward Network (MixCFN) [39], and Convolutional Gated Linear Unit (ConvGLU) [40]. The quantitative comparison is presented in Table V. Compared to SFFN, MixCFN, and ConvGLU, our HSSFGN achieved the best performance in terms of accuracy, parameter efficiency, and computational cost. Although DFN's accuracy is slightly higher than that of HSSFGN by 0.43%, it incurs an additional 53% in parameters and 27% in computational cost. Furthermore, we further explore the components of HSSFGN in Table VI. The results demonstrate that both the gated mechanism paradigm and the multi-scale convolution in the gated branch are indispensable to HSSFGN. The gated mechanism paradigm effectively promotes information flow, preventing blockage and redundancy. Meanwhile, the multi-scale convolution in the gated branch extracts multi-scale expert information across channels, working synergistically with the gated mechanism to enhance feature representation.

IV. Conclusion

This paper introduces an adaptive efficient sparse Transformer for food recognition, named Fraesormer. To adaptively model global long-range dependencies while enhancing the sparsity of the neural network, reducing redundant information, and minimizing computational burden, we developed an Adaptive Top-k Sparse Partial Attention to replace standard self-attention. Furthermore, we implemented an efficient multi-scale feedforward network based on a gating mechanism to further improve feature representation capabilities. Extensive experimental results indicate that Fraesormer achieves a favorable balance between performance and computational efficiency.

References

[1] Nidhi Rajesh Mavani, Jarinah Mohd Ali, Suhaili Othman, et al., "Application of artificial intelligence in food industry—a guideline," *Food Engineering Reviews*, 2022.

[2] Rebecca G Boswell, Wendy Sun, Shosuke Suzuki, and Hedy Kober, "Training in cognitive strategies reduces eating and improves food choice," *Proceedings of the National Academy of Sciences*, 2018.

[3] Wenjing Shao, Weiqing Min, et al., "Vision-based food nutrition estimation via rgb-d fusion network," *Food Chemistry*, 2023.

[4] Yancun Yang, Weiqing Min, et al., "Lightweight food recognition via aggregation block and feature encoding," *ACM Transactions on Multimedia Computing, Communications and Applications*, 2024.

[5] Zhiling Wang, Weiqing Min, Zhuo Li, et al., "Ingredient-guided region discovery and relationship modeling for food category-ingredient prediction," *TIP*, 2022.

[6] Chairi Kiourt, George Pavlidis, and Stella Markantonatou, "Deep learning approaches in food recognition," *Machine learning paradigms: advances in deep learning-based technological applications*, 2020.

[7] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool, "Food-101 – mining discriminative components with random forests," in *ECCV*, 2014.

[8] Jingjing Chen and Chong-wah Ngo, "Deep-based ingredient recognition for cooking recipe retrieval," in *ACM MM*, 2016.

[9] Yoshiyuki Kawano and Keiji Yanai, "Foodcam-256: A large-scale real-time mobile food recognitionsystem employing high-dimensional features and compression of classifier weights," in *ACM MM*, 2014.

[10] Jianing Qiu, Frank Po Wen Lo, Yingnan Sun, Siyao Wang, and Benny Lo, "Mining discriminative food regions for accurate food recognition," in *BMVC*, 2019.

[11] Javier Ródenas et al., "Learning multi-subset of classes for fine-grained food recognition," in *Proceedings of the International Workshop on Multimedia Assisted Dietary Management*, 2022.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *ICLR*, 2021.

[13] Yanyu Li, Ju Hu, Yang Wen, Georgios Evangelidis, et al., "Rethinking vision transformers for mobilenet size and speed," in *ICCV*, 2023.

[14] Pavan Kumar Anasosalu Vasu et al., "Fastvit: A fast hybrid vision transformer using structural reparameterization," in *ICCV*, 2023.

[15] Simone Bianco, Marco Buzzelli, et al., "Food recognition with visual transformers," in *Proceedings of the International Conference on Consumer Electronics-Berlin*, 2023.

[16] Hila Chefer, Shir Gur, and Lior Wolf, "Transformer interpretability beyond attention visualization," in *CVPR*, 2021.

[17] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou, "Training data-efficient image transformers & distillation through attention," in *ICML*, 2021.

[18] Sachin Mehta and Mohammad Rastegari, "Separable self-attention for mobile vision transformers," *arXiv preprint arXiv:2206.02680*, 2022.

[19] Pichao Wang, Xue Wang, et al., "Kvt: k-nn attention for boosting vision transformers," in *ECCV*, 2022.

[20] Yuxin Liu, Weiqing Min, Shuqiang Jiang, and Yong Rui, "Convolution-enhanced bi-branch adaptive transformer with cross-task interaction for food category and ingredient recognition," *TIP*, 2024.

[21] Qihang Fan, Huaibo Huang, Mingrui Chen, Hongmin Liu, and Ran He, "Rmt: Retentive networks meet vision transformers," in *CVPR*, 2024.

[22] Jianyuan Guo, Kai Han, Han Wu, Yehui Tang, Xinghao Chen, Yunhe Wang, and Chang Xu, "Cmt: Convolutional neural networks meet vision transformers," in *CVPR*, 2022.

[23] Lei Zhu, Xinjiang Wang, et al., "Biformer: Vision transformer with bi-level routing attention," *CVPR*, 2023.

[24] Xiangxiang Chu, Zhi Tian, et al., "Conditional positional encodings for vision transformers," in *ICLR*, 2023.

[25] Syed Waqas Zamir, Aditya Arora, et al., "Restormer: Efficient transformer for high-resolution image restoration," in *CVPR*, 2022.

[26] Jiangning Zhang, Xiangtai Li, Jian Li, et al., "Rethinking mobile block for efficient attention-based models," in *ICCV*, 2023.

[27] Ao Wang, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding, "Repvit: Revisiting mobile cnn from vit perspective," in *CVPR*, 2024.

[28] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, et al., "Searching for mobilenetv3," in *ICCV*, 2019.

[29] Abdelrahman Shaker, Muhammad Maaz, et al., "Swiftformer: Efficient additive attention for transformer-based real-time mobile vision applications," in *ICCV*, 2023.

[30] Jierun Chen, Shiu-hong Kao, Hao He, et al., "Run, don't walk: chasing higher flops for faster neural networks," in *CVPR*, 2023.

[31] Junting Pan, Adrian Bulat, et al., "Edgevits: Competing light-weight cnns on mobile devices with vision transformers," in *ECCV*, 2022.

[32] Pavan Kumar Anasosalu Vasu, James Gabriel, et al., "Mobileone: An improved one millisecond mobile backbone," in *CVPR*, 2023.

[33] Yehui Tang, Kai Han, et al., "Ghostnetv2: Enhance cheap operation with long-range attention," *NeurIPS*, 2022.

[34] Seokju Yun and Youngmin Ro, "Shvit: Single-head vision transformer with memory efficient macro design," in *CVPR*, 2024.

[35] Mingxing Tan and Quoc Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *ICML*, 2019.

[36] Weihao Yu, Mi Luo, Pan Zhou, et al., "Metaformer is actually what you need for vision," in *CVPR*, 2022.

[37] Yun Liu, Yu-Huan Wu, Guolei Sun, et al., "Vision transformers with hierarchical attention," *Machine Intelligence Research*, 2024.

[38] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool, "Localvit: Bringing locality to vision transformers," *arXiv preprint*, 2021.

[39] Jiaqi Gu, Hyoukjun Kwon, et al., "Multi-scale high-resolution vision transformer for semantic segmentation," in *CVPR*, 2022.

[40] Dai Shi, "Transnext: Robust foveal visual perception for vision transformers," in *CVPR*, 2024.