SMART-NIR: A Multi-Kernel Vision Transformer with Kolmogorov-Arnold Network Classifier for Near-Infrared Spectral Classification

Nguyen Thi Hoang Phuong[1], Phan Minh Nhat[2], Nguyen Van Hieu[2,*]
[1]Pham Van Dong University, 509 Phan Dinh Phung, Quang Ngai, Vietnam
[2]The University of Danang - University of Science and Technology, 54 Nguyen Luong Bang, Da Nang, Vietnam
E-mail: nthphuong@pdu.edu.vn, nhat0299@gmail.com, nvhieuqt@dut.udn.vn
*Correspoding Author

Keywords: Multi-kernel, transformer, Kolmogorov-Arnold networks, dual-MLP, near-infrared spectroscopy

Received: October 24, 2024

Near-infrared (NIR) spectroscopy is a prevalent analytical technique employed in classification and quality control across various domains. However, its efficacy is often constrained by the inherent complexity, high dimensionality, and nonlinear characteristics of spectral data. This study introduces SMART-NIR, a novel analytical framework designed to address these challenges. The proposed approach was evaluated on the SpectroFood dataset, which comprises samples represented by 240 wavelength features. SMART-NIR incorporates a multi-kernel feature extraction mechanism, a modified Vision Transformer architecture equipped with a Dual-MLP module, and Kolmogorov-Arnold Networks (KAN) to enhance classification accuracy. To accommodate variable-length inputs, zero-padding was employed, and model robustness was assessed through five-fold cross-validation. The framework achieved a classification accuracy of 99.24%, demonstrating an 8% improvement over a baseline Transformer and a 5% gain relative to a standard multilayer perceptron (MLP) classifier. Furthermore, the Dual-MLP architecture effectively reduces the number of parameters and floating-point operations (FLOPs) compared to conventional Transformer-based feed-forward networks. These findings underscore SMART-NIR's capability to model complex nonlinear relationships in spectral data, positioning it as a robust solution for real-time quality assessment and analysis in dynamic and noise-prone environments.

Povzetek: SMART-NIR, novi analitični okvir, združuje večjedrno (multi-kernel) ekstrakcijo značilnosti, izboljšan Vision Transformer (ViT) z Dual-MLP in Kolmogorov-Arnold Network (KAN) klasifikator za obdelavo podatkov blizu-infrardeče (NIR) spektroskopije. Okvir je dosegel odlične rezultate na naboru podatkov SpectroFood, kar potrjuje robustno modeliranje kompleksnih nelinearnih spektralnih podatkov.

1 Introduction

Near-infrared spectroscopy has become an essential analytical technique in many fields, including food science, pharmaceuticals, chemistry, and agriculture. Its advantages include the ability to perform rapid, non-invasive, and non-destructive analysis of samples with low operating costs [1]. However, processing and analyzing NIR spectral data remain challenging due to its inherent complexity [2, 3]. These main difficulties include the wide wavelength range from 750 to 2500 nm, the non-linear wavelength relationships, overlapping, and the susceptibility of NIR spectra to environmental factors such as temperature and humidity [3, 4, 5].

Over the years, many machine learning methods have been applied to address the challenges in NIR spectra processing, focusing on two main directions: classification and regression. Traditional, widely adopted methods, such as Principal Component Analysis, Soft Independent Modeling of Class Analogy, Linear Discriminant Analysis, and Partial Least Squares Discriminant Analysis, have still achieved outstanding results for data dimensionality and categorizing data in the past three years [6, 7, 8, 9]. Meanwhile, other methods such as Support Vector Machine, Decision Tree, Random Forest, Artificial Neural Network, and Extreme Learning Machine [7, 10, 11, 12, 13] have been effective in handling nonlinear data relationships. However, traditional machine learning is more suitable for limited data and computational resources. In addition, finding the best preprocessing and feature extraction methods for each dataset depends mainly on the expert's experience.

Recently, deep learning methods have been developed vigorously and have been successfully applied to both NIR spectral regression and classification [14, 15, 16]. In particular, Convolutional Neural Networks (CNN) are effective in extracting local features from spectral data. In addition, Recurrent Neural Networks (RNN) [17] and its variants, such as Long Short-Term Memory Networks (LSTM) [18, 19], have been used to process spectral data. Other architectures, such as Autoencoders (AE) [20, 21] and Generative Adversarial Networks (GAN) [22], have also been tested in learning representations and modeling NIR spec-

tral data. Although these methods have succeeded, they have some significant limitations. CNNs excel at extracting local features but struggle with global features in spectral data. RNNs and LSTMs can handle time series but face vanishing or exploding gradients with long sequences. While AEs and GANs are effective for representation learning and pattern generation, they encounter difficulties when dealing with the complexity of spectral information. Therefore, towards efficiently handling both short-term and long-term relationships in data, while being able to generalize the complex and diverse characteristics of spectral data as mentioned in the previous section, more suitable methods still need to be developed.

Transformer, introduced by Google in 2017 [23], has quickly become a pivotal architecture in natural language processing and has subsequently been expanded to many other areas, including NIR spectroscopy processing. Transformer combines a multi-head self-attention (MHSA) mechanism and a multi-layer perceptron network (MLP) in an encoder-decoder structure, allowing simultaneous learning of global correlations and local nonlinear representations, which can efficiently process long sequences and capture complex relationships in the data, outperforming traditional methods such as CNN and RNN [24, 25] in processing sequence data. For instance, the Swin Transformer [24] effectively predicted soil properties directly from raw NIR spectra, outperforming CNN-based approaches. In [25], the proposed improved Transformer model combining spatial and channel-level representations of fNIRS signals, using a convolutional layer and a new preprocessing module, achieved the highest accuracy on three open datasets, with 75.49% on the most complex 3-class classification problem (surpassing CNN by 2.6% and LSTM by 13.55%), and an average accuracy of 78.22% in the single-subject test (surpassing CNN by 4.75% and LSTM by 11.33%).

Vision Transformer (ViT), a variant of Transformer specifically designed for computer vision tasks, has demonstrated significant potential in processing NIR spectral data. ViT views the NIR spectrum as a series of "patches" and applies a self-attention mechanism to learn the relationships between these patches. In [26], ViT and CNN performed better by efficiently processing local and global information from spectral data encoded into 2D images. Similarly, the CT-Net model in [27] recently combined CNN and ViT with a novel signal representation (separate 1D CNN for HbO and HbR), achieving 98.05% and 77.61% accuracy on two fNIRS datasets, outperforming other deep learning models. Considering the research on processing pseudo-spectral images generated from spectra on both traditional Transformer and ViT architectures, this is a relatively new but promising approach. However, it still has limitations in generalizability on different objects, has many hyperparameters to tune, faces challenges of high computational complexity, especially in the self-attention mechanism, as well as the need to reduce the number of unwanted features from the data to be able to deploy more widely in practice. Moreover, especially with the trend of using handheld spectroscopic devices as alternatives to laboratory benchtop instruments, NIR spectral data fluctuates sensitively with environmental factors, requiring greater flexibility and adaptability from machine learning models.

Introduced in April 2024, Kolmogorov-Arnold Networks (KAN) emerged as a promising neural network architecture considered a viable alternative to MLP [28]. Based on the Kolmogorov-Arnold representation theorem, KAN uses learnable activation functions on edges instead of fixed weights or activation functions like MLP. This method enables more efficient modeling of complex nonlinear relationships and enhances interpretability through symbolic regression. It opens up a new research direction for combining the advantages of ViT and KAN in NIR spectrum processing. As far as we know, this study is unprecedented. In this study, we propose a new SMART-NIR model with the main architecture including:

1. The multi-kernel block integrates kernels from 4×1 to 32×1 in parallel to diversify feature extraction. The concatenated outputs enable multi-scale feature representation, balancing local details and global context, which helps reduce noise and capture broader trends. For classification tasks, this design enhances discriminative power and improves overall accuracy by allowing the model to better adapt to varying patterns in the input data.

2. Transformer Encoder is improved with Dual-MLP, replacing the traditional MLP by splitting the input in half and processing it in parallel through two separate branches, each consisting of a linear layer, GELU activation function and a linear layer, allowing the model to simultaneously learn linear and non-linear representations, enhancing the ability to capture complex patterns and improving performance without significantly increasing computational costs.

3. KAN is integrated as a complete replacement for traditional MLP classifiers in the classification head. This approach enables the flexible approximation of complex functions using a minimal number of parameters, without increasing the overall network size. On NIR classification tasks, KAN outperforms traditional MLP classifiers by achieving higher accuracy and improved generalization across diverse and noisy data. Its ability to model non-linear relationships more effectively makes it particularly suited for complex spectral variations common in NIR data.

The SMART-NIR aims to build a robust self-attention multi-kernel adaptive representation model for NIR spectral data. This method significantly improves the performance in NIR spectral classification tasks while improving generalization and computational efficiency. In the following sections, we will present the details of the proposed architecture, experimental method, and performance evaluation of SMART-NIR on a diverse food dataset.

2 Related work

2.1 MLP and KAN

Over the past decades, MLPs have served as a fundamental architecture in deep learning. Even the prominent Vision Transformer architecture, while not a direct extension of MLPs, incorporates MLP components into its architecture. MLP, based on the universal approximation theorem, consists of multiple fully connected layers with fixed activation functions [28]. For an L-layer MLP, with input vector $\mathbf{x} \in \mathbb{R}^d$, the output can be represented as:

$$\text{MLP}(\mathbf{x}) = (\mathbf{W}_{L-1} \circ \sigma \circ \mathbf{W}_{L-2} \circ \sigma \circ \cdots \circ \mathbf{W}_1 \circ \sigma \circ \mathbf{W}_0)\mathbf{x}$$

where W_i is the weight matrix for the i^{th} layer respectively, and σ is a non-linear activation function, typically ReLU or GELU, which is usually the same for all layers.

However, KAN has emerged as a promising alternative, offering some significant advantages over traditional MLP for specific tasks. Unlike MLP, KAN is based on the Kolmogorov-Arnold representation theorem and shows better adaptive flexibility. It replaces linear weights and fixed activation functions with learnable univariate functions B-spline on network edges [28], adapting dynamically during training to capture complex data patterns.

A single KAN layer is represented: $\Phi = \{\phi_{q,p}\}$, where $p = 1, 2, ..., n_{in}$, $q = 1, 2, ..., n_{out}$ and $\phi_{q,p}$ are parametrized functions with learnable coefficients. This structure enables KAN to model non-linear relationships in data effectively. An L-layer KAN network is a composition of L such layers:

$$\text{KAN(x)} = (\Phi_{L-1} \circ \Phi_{L-2} \circ ... \circ \Phi_0)\mathbf{x}$$

In [28], it was shown that with fewer parameters and fewer layers, KAN outperforms MLP in learning complex data structures. An essential advantage of KAN is its ability to automatically discover optimal structures through sparsification and pruning, which helps to learn important features and remove redundant information [29]. In addition, KAN's symbolic regression explainability allows for understanding the model's decision-making process, which is difficult with conventional deep learning architectures.

KAN has demonstrated outstanding performance in many applications. They've outperformed MLP, GRU, and LSTM in time series tasks with fewer parameters [30, 31], while enhancing interpretability. Additionally, KANs enhance the interpretability of results by applying symbolic regression on B-spline activation functions [32]. For hyperspectral image classification, Wav-KAN and SpectralKAN excelled on Salinas, Pavia, and Indian Pines datasets [33]. Wav-KAN achieved 92.62% accuracy and 0.9157 kappa coefficient overall, outperforming Spline-KAN (89.85%, 0.8793) and MLP (77.69%, 0.7119). On Indian Pines, Wav-KAN (85.54%, 0.8348) significantly surpassed Spline-KAN (77.31%, 0.7395) and MLP (35.13%, 0.2984). Nevertheless, in a separate study [29] directly comparing KAN and MLP across various tasks (computer vision, natural language processing, and audio processing) while maintaining identical parameter settings or FLOPs, KAN has yet to exhibit a clear superiority.

The outcomes undeniably showcase KAN's immense promise. Nonetheless, substantial research is imperative to optimize its framework and unlock its full potential in a variety of machine learning applications.

2.2 Deep learning approach for NIR spectral classification

Deep learning has emerged as a transformative paradigm across signal processing domains, including NIR spectroscopic signal, offering robust methodologies for addressing complex classification and regression tasks [34, 35]. Unlike conventional machine learning techniques that typically depend on manual feature engineering, deep learning architectures can autonomously learn hierarchical feature representations through successive layers of nonlinear transformations applied directly to raw spectral data. These multilayered structures, commonly referred to as deep architectures, facilitate end-to-end learning and have demonstrated substantial efficacy in approximating complex nonlinear functions, modeling high-dimensional datasets, and achieving strong generalization performance even when trained on relatively limited samples [36, 35].

Within the context of NIR spectroscopy, various deep learning models, such as CNNs, RNNs, autoencoders, and hybrid frameworks, have exhibited significant potential [35]. These models are particularly adept at capturing spatial and temporal dependencies intrinsic to spectral data, thereby enhancing both classification accuracy and model robustness. CNNs, in particular, have been extensively employed due to their proficiency in extracting localized, shift-invariant features from spectral sequences [37]. Enhanced variants, including one-dimensional CNNs [38, 39], CNN-RNN hybrid models [40], and Transformer-based models [41] have further improved performance in scenarios involving dynamic spectral variations or temporal dependencies.

Empirical studies have demonstrated the diverse applicability and advantages of deep learning in NIR spectral analysis. Autoencoder-based models, for instance, have been successfully employed in the non-destructive and rapid analysis of bright-blue pigments in cosmetic creams, achieving high predictive accuracy while reducing computational overhead [42]. CNN-based transfer learning approaches have also proven effective in pharmaceutical applications, notably in drug classification tasks, where they attained high accuracy with minimal labeled data [43]. In agricultural and food quality control domains, CNNs have been deployed to assess product freshness [44], predict soil properties [37, 45], and classify fruits and dried produce [39, 46], consistently outperforming traditional chemometric approaches in predictive capability.

Beyond standard CNN architectures, more specialized

and hybrid deep learning models have been developed to further optimize learning processes and improve performance. These include stacked autoencoders (SAEs) [36], variational autoencoders (VAEs) [47], and local receptive field–based extreme learning machines (LRF–ELMs) [48, 49]. Additionally, composite models such as convolutional neural networks combined with gravitational reservoir computing (CNN-GRC-ELM) [50] and CNNs integrated with decision trees [51] have been proposed to enhance robustness and interpretability.

Despite the evident benefits, the application of deep learning in NIR spectroscopy is not without challenges. Key limitations include the dependence on large, annotated datasets for effective training [52, 35], sensitivity to preprocessing methods [53, 54], and the substantial computational resources often required for model optimization and deployment [37]. Furthermore, in many studies, evaluation has been conducted on relatively small experimental datasets, with limited comparative analysis against alternative deep learning frameworks [38, 50]. Model interpretability also remains a critical concern, particularly for real-world industrial deployment, where transparency and explainability are essential [35]. Addressing these limitations through the development of interpretable architectures, hybrid learning strategies, and efficient transfer learning mechanisms is an important direction for future research.

In conclusion, deep learning offers a highly adaptable and potent framework for NIR spectral classification and analysis. Its ability to automatically extract complex, nonlinear patterns from spectral data enables its application across a wide range of sectors, including food safety, environmental monitoring, pharmaceutical quality control, and agricultural assessment. As deep learning models continue to evolve and larger, more diverse datasets become available, their role in advancing the capabilities of NIR spectroscopy is expected to become increasingly pivotal.

3 Methodology

3.1 Multi-kernel convolution analysis

The study explores how kernel dimensions affect feature extraction by analyzing the outputs of four convolutional layers. These layers process a 512×1 signal using kernels of increasing size, spanning from 4×1 to 32×1. The results demonstrate that the kernel size is a significant factor in determining the effectiveness of feature learning.

Employing a small kernel size 4×1 within a convolutional layer restricts the receptive field to proximate pixels, enabling the capture of fine-grained, localized features. Consequently, this configuration excels at discerning high-frequency components and short-term temporal patterns. However, the sensitivity to local information can render the model susceptible to noise interference, potentially leading to inaccurate representations, especially when processing data with high noise levels.

Larger kernel sizes, such as 8×1, 16×1, and 32×1, expand the convolutional receptive field, enabling the capture of broader contextual information. This configuration is particularly effective in extracting low-frequency components, long-range temporal dependencies, and amplitude variations within the input data. However, the computational overhead associated with large kernels can negatively impact model performance and efficiency. To mitigate this, a multi-scale approach incorporating both small and large kernels is often employed.

3.2 SMART-NIR architecture

The ViT architecture excels in capturing global features and offers scalable designs. However, traditional ViT-based methods for image classification often employ 3×3 convolutional projections during feature embedding, which may not optimally preserve crucial signal characteristics. To address this limitation and enhance global feature extraction, we propose a novel SMART-NIR model, that effectively captures a richer set of signal characteristics within each token. The overall architecture of SMART-NIR is presented in Figure 1.

3.2.1 Multi-kernel block

The input signal is represented as a vector $\mathbf{r}$ of length L. This vector is subsequently reshaped into a three-dimensional tensor $\mathbf{I}$ with dimensions $1 \times L \times 1$, where L is the signal length, fixed at 512 for this study. A parallel multi-kernel architecture is employed, consisting of four processing branches equipped with kernel sizes of 4×1, 8×1, 16×1, and 32×1, respectively. This configuration facilitates the extraction of features at multiple scales, as illustrated in Figure 1. A kernel $\mathbf{K}$ with dimensions $C_{\text{out}} \times K_w \times K_h$ is applied in the convolutional layer to extract features from the input data. Here, C_{out}, K_w, and K_h correspond to the number of output channels, kernel width, and kernel height, respectively. The operation incorporates padding P and stride S. The output tensor from this process can be computed according to the following equations:

$$
\begin{aligned}
O_1 &= \text{Conv}\left(\mathbf{I}, \mathbf{W}_1, \mathbf{K}_1 = (C_{\text{out}}, 4, 1), S, P_1 = (0, 0)\right), \\
O_2 &= \text{Conv}\left(\mathbf{I}, \mathbf{W}_2, \mathbf{K}_2 = (C_{\text{out}}, 8, 1), S, P_2 = (3, 0)\right), \\
O_3 &= \text{Conv}\left(\mathbf{I}, \mathbf{W}_3, \mathbf{K}_3 = (C_{\text{out}}, 16, 1), S, P_3 = (7, 0)\right), \\
O_4 &= \text{Conv}\left(\mathbf{I}, \mathbf{W}_4, \mathbf{K}_4 = (C_{\text{out}}, 32, 1), S, P_4 = (15, 0)\right).
\end{aligned}
$$

Here, $\mathbf{W}_i$ represents the weights of the convolutional operations, respectively, where $i \in \{1, 2, 3, 4\}$. A stride of $(4, 1)$ is applied consistently across all convolutions. The multi-kernel module produces a final output tensor, $\mathbf{X}_0$, by concatenating the output tensors from each convolutional layer ($\mathbf{O}_1$, $\mathbf{O}_2$, $\mathbf{O}_3$, and $\mathbf{O}_4$) along the channel dimension. This results in a tensor with a depth equal to the sum of the depths of the individual output tensors.

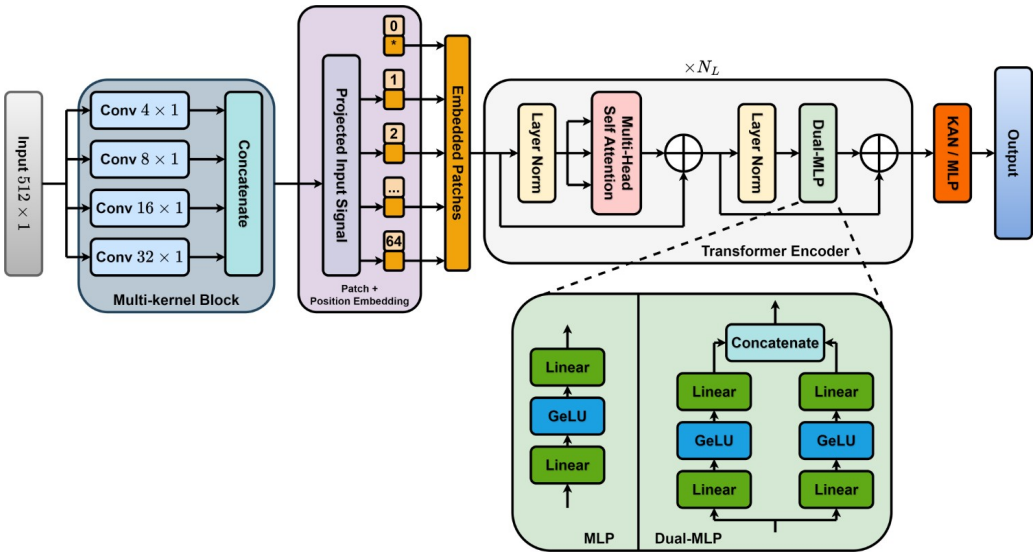

Figure 1: SMART-NIR architecture

3.2.2 Position embedding

The output of the multi-kernel block, denoted as $\mathbf{X}_0$, has dimensions $C_{\text{out}} \times H_{\text{out}} \times 1$, where C_{out} is the total number of output channels and H_{out} is the output height calculated as $H_{\text{out}} = (L - K + 2P)/S + 1$. A linear position embedding is generated by first transposing the output tensor, $\mathbf{X}_0$, and then flattening the resulting tensor along the spatial dimensions. This flattened output is then concatenated with the token $\mathbf{x}_{\text{cls}}$, followed by the addition of a learnable positional bias $\mathbf{E}_{\text{pos}}$ with dimensions $(H_{\text{out}} + 1) \times C_{\text{out}}$. The resulting output, $\mathbf{X}_1$, can be expressed as: $\mathbf{X}_1 = \text{Concat}(\mathbf{x}_{\text{cls}}, \mathbf{X}_0^T) + \mathbf{E}_{\text{pos}}$.

3.2.3 The transformer encoder

This module is designed to capture complex dependencies within the input sequence. It is composed of a stack of layers, each of which includes a multi-head self-attention mechanism followed by a feedforward neural network. This alternating structure enables the model to effectively process and understand the relationships between different elements of the input data. Layer normalization (LN) is utilized as a preprocessing step for each block, and skip connections help to mitigate the vanishing gradient problem.

Multi-Head Self-Attention: is a core building block of transformer architectures. Drawing on the principle of self-attention, it enables the model to create weighted representations of input features by accounting for their interdependencies within a sequence. Unlike traditional sequential processing, this approach allows the model to identify complex relationships and dependencies that might be difficult to detect in a linear sequence. This mechanism allows the model to capture complex relationships between elements within the input sequence, enhancing its ability to model long-range dependencies.

In this research, the input sequence $\mathbf{X}_1 \in \mathbb{R}^{(H_{\text{out}}+1) \times C_{\text{out}}}$ is employed. To compute the weighted sum of sequence elements, the generation of three key vectors is essential: the Query vector (**Q**), the Key vector (**K**), and the Value vector (**V**). The Query vector, **Q**, is derived by multiplying $\mathbf{X}_1$ by $\mathbf{W}_Q$: $\mathbf{Q} = \mathbf{X}_1\mathbf{W}_Q$. Subsequently, the Key vector, **K**, is created through the multiplication of $\mathbf{X}_1$ and $\mathbf{W}_K$: $\mathbf{K} = \mathbf{X}_1\mathbf{W}_K$. Finally, the Value vector, **V**, is obtained by multiplying $\mathbf{X}_1$ and $\mathbf{W}_V$: $\mathbf{V} = \mathbf{X}_1\mathbf{W}_V$. These three vectors, **Q**, **K**, and **V**, serve as the basis for further calculations.

The attention mechanism calculates the attention weight between each pair of tokens by computing the dot product of their corresponding query and key vectors. This result is then scaled by the inverse of the square root of the key vector's dimensionality to maintain numerical stability. The scaled dot products are then passed through a softmax function, producing a set of normalized attention weights that indicate the relative importance of each token in the sequence with respect to the others:

$$\text{SA}(\mathbf{X}_1) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{H_{\text{out}} + 1}}\right)\mathbf{V}$$

where $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ are trainable weight matrices.

MHSA extends the standard self-attention mechanism by performing multiple parallel attention computations, known as heads. Each head independently processes the input data through a distinct learned linear projection, enabling the capture of diverse relational features within the input. The concatenated outputs from all attention heads are subjected to a linear transformation, yielding the final MHSA output. This process can be formally expressed as: $\text{MHSA}(\mathbf{z}) = \text{Concat}(\text{SA}_1(\mathbf{X}_1), \text{SA}_2(\mathbf{X}_1), ..., \text{SA}_{N_h}(\mathbf{X}_1))$, where N_h denotes the number of heads.

Dual-MLP architecture is introduced as a method to decrease the computational demands of the Transformer model while maintaining or even improving its performance. By splitting the MLP input into two equal parts,

the model is enabled to learn diverse patterns more effectively while maintaining computational efficiency. This approach increases model capacity and flexibility without incurring significant overhead due to its parallel processing nature. The dual-branch architecture offers a specialized approach to data feature extraction, resulting in enhanced performance when handling intricate data patterns. The MHSA block's output tokens **z** are partitioned into two equal-sized components, **m** and **n**, by splitting them along their last axis. These parts are then subjected to separate linear transformations to form the Dual-MLP, as detailed below:

$$\text{Dual-MLP} = \text{Concat}(\mathbf{W}_{\text{fc4}}\text{GELU}(\mathbf{W}_{\text{fc}_2}\mathbf{m}), \mathbf{W}_{\text{fc3}}\text{GELU}(\mathbf{W}_{\text{fc1}}\mathbf{n}))$$

where $\mathbf{W}_{\text{fc1}}$, ..., $\mathbf{W}_{\text{fc4}}$ are the trainable parameters that define the linear transformations used in the four layers depicted in Figure 1.

3.2.4 Classifier

The task is performed using either KAN or MLP configured with two hidden layers with decreasing dimensionality (32, 16). In the case of KAN, as illustrated in the Figure 2, each connection between layers is not a simple linear weight as in traditional MLPs. Instead, KAN replaces the typical weight matrices with learnable univariate functions (denoted as red blocks in the diagram), which are applied to each input dimension individually.

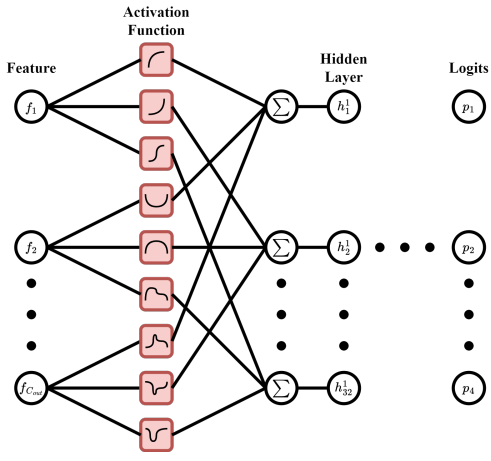

Figure 2: Integration of KAN into SMART-NIR

Each of these functions can learn complex nonlinear transformations, and the outputs are then summed to produce activations, similar to standard neural networks. This structure is inspired by the Kolmogorov–Arnold representation theorem, which states that any multivariate continuous function can be decomposed into a finite sum of univariate functions of linear combinations.

In the Figure 2, the input features **f** after the encoding process are passed through multiple univariate function blocks. The results are summed ($\sum$) to form the hidden layer **h**, which is then processed further to produce the final prediction logits **p**. Notably, KANs do not rely solely on linear matrix multiplications but instead leverage function-based transformations, which enhance the expressiveness of the model while preserving interpretability.

4 Experimental

4.1 Dataset

The dataset employed in this study was curated by Malounas Ioannis et al [55]. It comprises 1028 NIR spectral measurements derived from four distinct food categories: apples, broccoli, leeks, and mushrooms. Spectral data was collected across a wavelength range extending from 430 to 900 nm, with a resolution of 1.12 nm per pixel, resulting in a total of 240 wavelength dimensions. Figure 3 presents a visual representation of the sample size for each food category, while Figure 4 depicts the spectral measurements obtained for each category. A notable degree of uniformity was observed in the sample size across food types, with values ranging between 240 and 300 samples.

The spectral data, composed of 240 wavelength dimensions, was directly fed into the model. Given the model's architecture was designed for a 512-dimensional input, a discrepancy arose. To reconcile this dimensional disparity, zero-padding, a standard data preprocessing technique, was applied. This process involved appending zeros to both extremities of each spectrum, effectively expanding its dimensionality to 512. This manipulation not only ensures dimensional compatibility with the model input but also helps preserve edge information in spectral data, similar to its benefits in image processing. This reduces boundary effects and allows the model to better capture patterns across the full spectrum, enhancing classification performance [56].

4.2 Implementation details

The entire training and evaluation process was conducted on the Windows 10 operating system, utilizing an Intel® Xeon® Platinum 8470Q processor and an NVIDIA DGX A100 graphics card. The software employed Python version 3.10 and the Pytorch CUDA 12.1 framework.

The entire dataset was partitioned into 5 non-overlapping subsets using stratified 5-fold cross-validation, ensuring that the class distribution in each fold matched the overall target distribution. In each round of validation, one fold was used as the test set while the remaining four served as the training set. The evaluation metrics were averaged across the five folds to obtain a robust estimate of model performance. To ensure reproducibility, a fixed random seed (43) was used during data splitting and throughout training. To mitigate class imbalance within each training fold, class balancing was applied via sampling, allowing the model to receive an equitable representation of all classes during training.

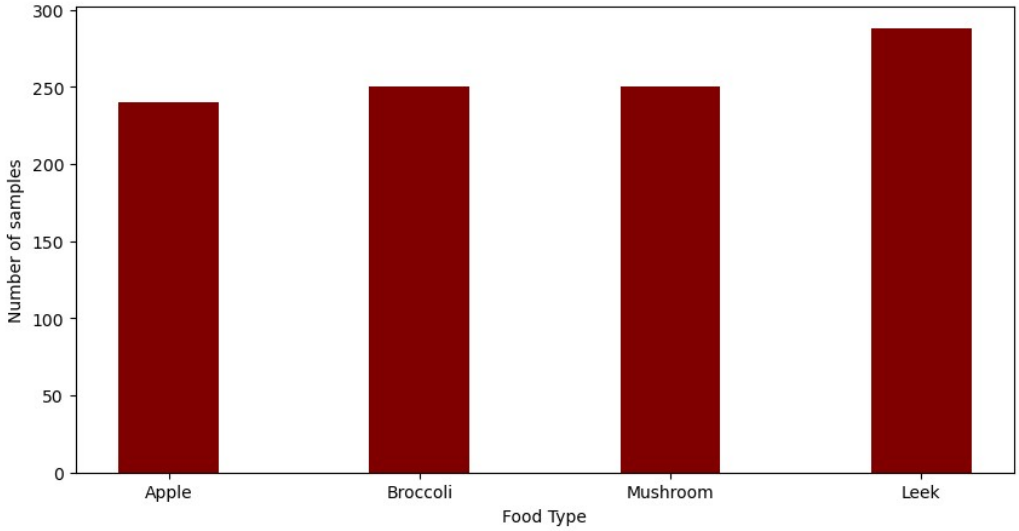

Figure 3: Number of samples for each food category

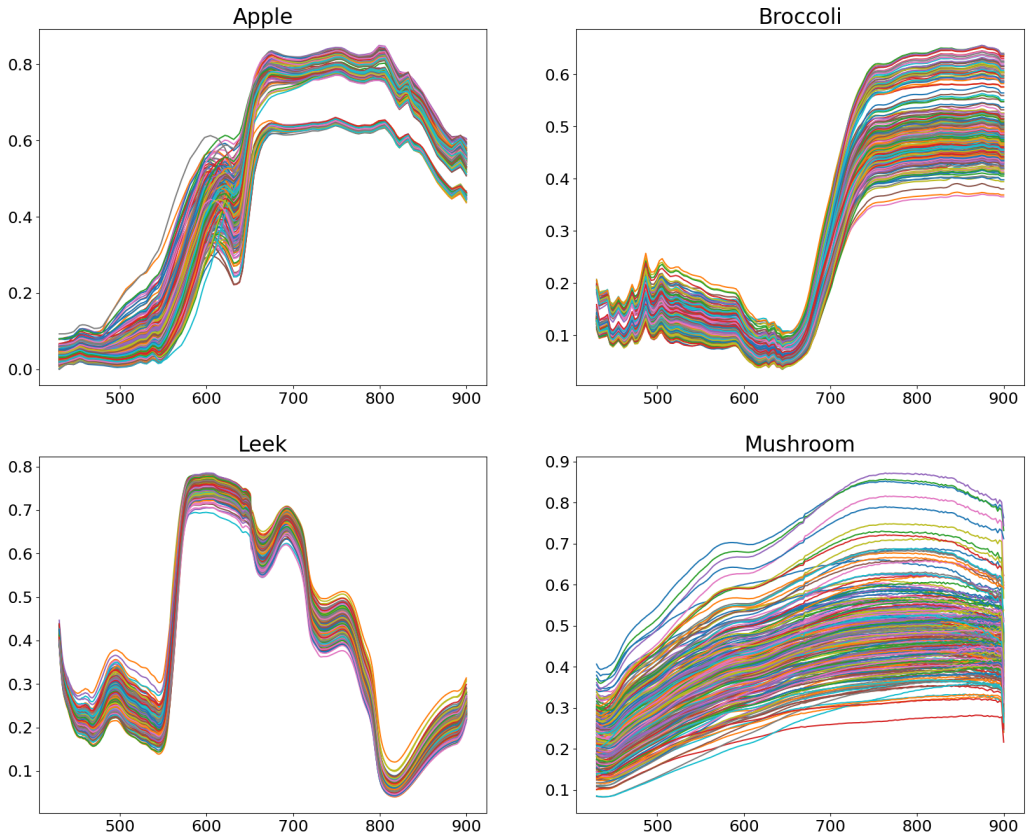

Figure 4: NIR spectral measurements obtained for each food category

To assess the model's robustness under noisy conditions, additive Gaussian noise was introduced to the test set in each fold. Specifically, zero-mean Gaussian noise with a random standard deviation sampled from the range $[0.1, 0.5]$ (with a step of 0.1) was applied to each wavelength to simulate sensor noise and real-world variation in spectral measurements.

All models were trained heuristically for 200 epochs using Stochastic Gradient Descent (SGD) with an initial learning rate of 10^{-5}. The learning rate was dynamically adjusted using the Cosine Annealing with Warm-Up schedule. The training objective was defined by the Multi-Class Cross-Entropy loss, appropriate for the classification task.

KAN classifier integrated in SMART-NIR is trained us-

ing a gradient-based optimization approach similar to traditional neural networks, but with key differences in parameterization and initialization. Each connection in a KAN is associated with a learnable univariate function, using spline functions such as cubic B-splines or piecewise polynomials, rather than a scalar weight. These univariate functions are typically initialized to behave like identity functions (e.g., $f(x) \approx x$) to ensure that the network starts in a stable, near-linear regime. The coefficients or control points of the spline functions are the trainable parameters in the model.

4.3 Evaluation

4.3.1 Quantitative results

To evaluate the effectiveness of the proposed method, we compared it against several well-established machine learning and deep learning approaches that are commonly used in spectral analysis. These include traditional models like MLP [57], KAN [29], and Random Forests, as well as advanced deep learning architectures such as 1D CNN [58], hybrid CNN-LSTM [59], and Transformer [23], which serve as a strong baseline in sequence modeling tasks.

The experimental results, presented in Table 1, clearly demonstrate the superior performance of SMART-NIR across all evaluation metrics. While conventional models such as MLP, KAN, and Random Forest achieved classification accuracies in the range of approximately 81–82%, deep learning models like CNN (86.81%), CNN-LSTM (87.63%), and baseline Transformer (91.29%) showed improved performance. However, SMART-NIR significantly outperformed all other methods, achieving an impressive 99.01% accuracy. This represents nearly an 8% improvement over the Transformer baseline. Moreover, SMART-NIR achieved consistently high Precision, Recall, and F1-Score values (all around 0.986), indicating not only high accuracy but also balanced and robust classification performance.

These findings highlight not only the technical advantages of SMART-NIR, particularly under noisy data conditions (simulated by adding noise to test robustness), but also its practical potential in real-world NIR-based applications. Therefore, SMART-NIR is not only a methodological advancement but also a meaningful contribution toward improving the reliability and accuracy of NIR-based classification systems.

4.3.2 Ablation study

An examination of the influence exerted by hyperparameters upon the performance of SMART-NIR is undertaken. Specifically, attention is directed towards pivotal hyperparameters such as kernel dimensions within the Multi Kernel and the quantity of layers and heads composing the transformer encoder. A series of comparative experiments is conducted with the objective of analysing the impact of variations in these hyperparameters upon SMART-NIR performance.

An investigation into kernel size was undertaken to ascertain the optimal multi-kernel configuration for the SMART-NIR model. Input and output channel dimensions were maintained at constant values of C_{in} and C_{out}, respectively. Consequently, the modulation of kernel size was prioritized to optimize the multi-kernel structure. Both single and multi-kernel configurations were subjected to evaluation, as detailed in Table 2. A gradual increment in accuracy was observed as kernel size was augmented. Based on these empirical findings, a multi-kernel configuration was implemented to augment feature extraction from the input spectra. Moreover, the substitution of the MLP classifier with KAN resulted in a substantial accuracy enhancement of approximately 5%, not only for the multi-kernel model but also for the single-kernel counterpart.

To further evaluate model robustness, noise was added to the input spectra to simulate real-world perturbations. As shown in Table 2, although a slight decrease in accuracy was observed across all configurations, the performance drop was not substantial. This can be attributed to the inherent separability of NIR spectral patterns between different classes, which provides a degree of resilience against input noise.

Notably, the multi-kernel configurations continued to outperform single-kernel ones even in noisy conditions, suggesting that multi-scale feature extraction contributes positively to robustness. Among the classifiers, the KAN-based models retained high accuracy and stable precision, recall, and F1-score values, confirming their superior generalization ability. The multi-kernel KAN model remained the top performer under noise, achieving 99.01% accuracy, indicating that SMART-NIR maintains strong classification capabilities even when faced with spectral variations.

An analysis of the number of layers (N_L) and number of heads (N_h) was conducted to optimize the transformer architecture within the SMART-NIR. To maintain a consistent model complexity, the relationship between the number of parameters and the number of floating-point operations (FLOPs) was considered.

The input features are projected using multiple convolution kernels of sizes $k \in \{4, 8, 16, 32\}$, each producing an output with C_{out} channels. Each convolution contributes $C_{in} \times C_{out} \times k$ weights and C_{out} biases. Summing over all kernel sizes:

$$\begin{aligned} P_{MK} &= \sum_{k \in \{4;8;16;32\}} C_{in} \times C_{out} \times k + C_{out} \\ &= 64 \times C_{out} \end{aligned}$$

Learnable positional encodings are added to the patch embeddings. Each of the N_p patches is associated with a vector of size $4C_{out}$, resulting in:

$$P_{PE} = N_p \times 4C_{out} = 256 \times C_{out}$$

The encoder consists of N_L layers, each with N_h attention heads. The parameters include the query, key, value,

Table 1: Comparison performance of SMART-NIR with different classification methods (noise adding)

Method	Accuracy	Precision	Recall	F1-Score
MLP (Single)	81.52 ± 0.41	0.813 ± 0.031	0.802 ± 0.029	0.807 ± 0.030
KAN (Single)	81.78 ± 0.39	0.784 ± 0.047	0.811 ± 0.056	0.797 ± 0.051
Random Forest	82.27 ± 0.42	0.812 ± 0.041	0.819 ± 0.049	0.815 ± 0.045
CNN (1D)	86.81 ± 0.48	0.861 ± 0.042	0.862 ± 0.041	0.861 ± 0.041
CNN-LSTM	87.63 ± 0.42	0.876 ± 0.031	0.873 ± 0.020	0.874 ± 0.024
Transformer (baseline)	91.29 ± 0.52	0.911 ± 0.019	0.919 ± 0.020	0.915 ± 0.019
SMART-NIR	$\mathbf{99.01 \pm 0.36}$	$\mathbf{0.987 \pm 0.026}$	$\mathbf{0.986 \pm 0.025}$	$\mathbf{0.986 \pm 0.025}$

Table 2: Comparison of the performance of the model using Dual-MLP, two types of classifiers, and different kernel sizes with $N_L = 4$, $N_h = 4$, and $C_{out} = 64$.

Classifier	Kernel Size	Accuracy	Precision	Recall	F1-Score
W/o Noise Adding					
MLP	4×1	91.21 ± 0.85	0.907 ± 0.025	0.905 ± 0.027	0.906 ± 0.024
	8×1	92.84 ± 0.78	0.857 ± 0.030	0.924 ± 0.026	0.889 ± 0.027
	16×1	93.63 ± 0.75	0.930 ± 0.022	0.864 ± 0.028	0.897 ± 0.025
	32×1	93.87 ± 0.70	0.941 ± 0.020	0.884 ± 0.024	0.911 ± 0.022
	Multi-kernel	94.63 ± 0.68	0.933 ± 0.018	0.938 ± 0.019	0.935 ± 0.017
KAN	4×1	98.33 ± 0.45	0.980 ± 0.026	0.971 ± 0.027	0.976 ± 0.026
	8×1	99.11 ± 0.35	0.954 ± 0.027	0.956 ± 0.025	0.955 ± 0.025
	16×1	99.16 ± 0.30	0.987 ± 0.024	0.980 ± 0.024	0.983 ± 0.024
	32×1	99.18 ± 0.28	0.973 ± 0.026	0.997 ± 0.023	0.985 ± 0.024
	Multi-kernel	$\mathbf{99.24 \pm 0.32}$	$\mathbf{0.993 \pm 0.025}$	$\mathbf{0.991 \pm 0.024}$	$\mathbf{0.992 \pm 0.024}$
Noise Adding					
MLP	4×1	89.03 ± 0.92	0.884 ± 0.030	0.881 ± 0.032	0.882 ± 0.028
	8×1	90.45 ± 0.88	0.836 ± 0.033	0.904 ± 0.029	0.868 ± 0.030
	16×1	91.28 ± 0.85	0.908 ± 0.028	0.842 ± 0.030	0.874 ± 0.029
	32×1	91.54 ± 0.81	0.917 ± 0.025	0.862 ± 0.027	0.889 ± 0.026
	Multi-kernel	92.31 ± 0.76	0.911 ± 0.023	0.917 ± 0.022	0.914 ± 0.021
KAN	4×1	97.45 ± 0.52	0.972 ± 0.029	0.963 ± 0.028	0.967 ± 0.028
	8×1	98.62 ± 0.43	0.946 ± 0.030	0.949 ± 0.029	0.947 ± 0.029
	16×1	98.79 ± 0.39	0.980 ± 0.027	0.973 ± 0.026	0.976 ± 0.026
	32×1	98.85 ± 0.37	0.967 ± 0.028	0.991 ± 0.025	0.979 ± 0.026
	Multi-kernel	$\mathbf{99.01 \pm 0.36}$	$\mathbf{0.987 \pm 0.026}$	$\mathbf{0.986 \pm 0.025}$	$\mathbf{0.986 \pm 0.025}$

and output projections (each with $C_{out}^2 + C_{out}$ parameters), and a Dual-MLP with two parallel branches, each consisting of two linear layers: the first projects to $\frac{H_{hidden}}{2}$ and the second projects back to $\frac{C_{out}}{2}$. The total encoder parameters are given by:

$$P_{Enc} = N_L \times \Big(N_h \times \Big(4 \times \big(C_{out}^2 + C_{out}\big) + 4 \times \big(\frac{C_{out}}{2} \times \frac{H_{hidden}}{2}\big)\Big)\Big)$$

The classification head includes a projection from C_{out} to 32- and 16-dimensional space, followed by a final classification layer projecting to N_{class} classes:

$$P_{cls} = C_{out} \times 32 + 32 \times 16 + 16 \times N_{class}$$

The total number of parameters P_{total} in the proposed SMART-NIR is computed as the sum of parameters across four main components: multi-kernel convolutional projection (P_{MK}), positional encoding (P_{PE}), Transformer encoder layers (P_{Enc}), and the final classification head (P_{cls}):

$$P_{total} = P_{MK} + P_{PE} + P_{Enc} + P_{cls}$$

This study investigates model variations within a parameter range spanning from 100K to 1000K with $H_{hidden} = 128$, intending to facilitate a comprehensive analysis of diverse model scales while acknowledging computational limitations. The results presented in Table 3 indicate that substantial improvements in accuracy are achieved through the augmentation of N_L, N_h, and C_{out}. These empirical findings suggest that the selection of $N_L = 6$, $N_h = 6$ and $C_{out} = 64$ yields optimal accuracy.

A comparative analysis of computational complexity and accuracy was undertaken between the proposed Dual-MLP and MLP components within the transformer encoder, as summarised in Table 4. The Dual-MLP mod-

Table 3: Comparison performance of SMART-NIR with different hyperparameters using Dual-MLP and KAN as classifiers (noise adding).

N_L	N_h	C_{out}	**Size (params)**	**FLOPs (M)**	**Accuracy (%)**
4	4	32	145K	12.1	98.91 ± 0.42
4	4	64	420K	41.9	98.95 ± 0.40
4	6	32	216K	12.1	98.92 ± 0.39
4	6	64	619K	41.9	98.96 ± 0.37
6	4	32	212K	14.2	98.93 ± 0.38
6	6	64	917K	47.2	99.01 ± 0.36

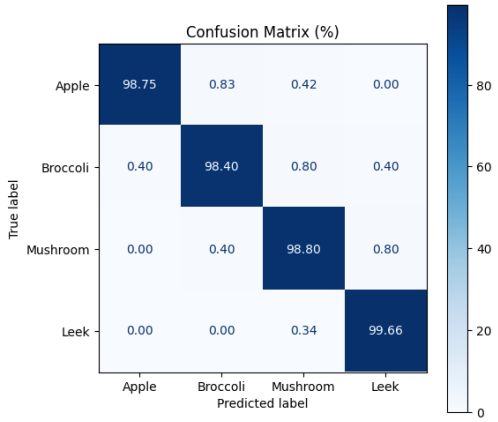

Figure 5: Confusion matrix showing the classification performance of Fold 2

ule attained the highest accuracy of 99.01% while simultaneously exhibiting an apprximate 25% reduction in both floating-point operations (FLOPs) and parameter count relative to the MLP. The Dual-MLP incorporates a linear projection of half the original MLP size, resulting in an approximate 25% reduction in parameters compared to the MLP. Dual-MLP demonstrated notably reduced computational demands compared to standard MLP, as measured by FLOPs. These results strongly support the conclusion that Dual-MLP is the optimal feed-forward choice for Transformer architectures.

As shown in Table 5, the model demonstrates a substantial performance advantage over the conventional Transformer model, achieving an accuracy increase of nearly 8%. Table 6 provides a detailed breakdown of the 5-fold cross-validation results for the SMART-NIR model, which was evaluated using the optimal hyperparameters. The confusion matrix in Figure 5 illustrates an example of the classification result from one fold.

Table 7 presents a comparison of inference time between SMART-NIR and a traditional Transformer architecture across different configurations of the number of layers (N_L) and attention heads (N_h). The results clearly demonstrate that SMART-NIR consistently achieves significantly lower inference times than the Transformer in all tested settings. For instance, with $N_L = 4$ and $N_h = 4$, SMART-NIR requires only 198.4 ± 14.7 ms, whereas the Transformer takes 235.2 ± 12.9 ms. Even in more complex settings such as $N_L = 6$ and $N_h = 6$, SMART-NIR maintains its efficiency with 293.4 ± 15.3 ms, compared to 377.9 ± 15.2 ms for the Transformer. These results highlight SMART-NIR's computational efficiency and suggest its strong potential for real-time or latency-sensitive applications where fast inference is critical.

Table 8 presents the evaluation results of the SMART-NIR model using different stride values. It is evident that as the stride increases, the model's performance consistently declines across all metrics, including Accuracy, Precision, Recall, and F1-Score.

Specifically, with a stride of 1, the model achieves the best performance, attaining an Accuracy of 99.01% and high Precision, Recall, and F1-Score values (all around 0.986–0.987). When the stride increases to 2, these metrics decrease moderately, and with a stride of 3, the decline becomes more pronounced (Accuracy drops to 97.80%, and F1-Score to 0.957).

This trend can be attributed to the effect of stride on feature extraction from the NIR spectral data. A smaller stride allows the model to capture more fine-grained spectral information, which is crucial for distinguishing subtle differences between classes. In contrast, a larger stride reduces the resolution of the input features, potentially omitting informative patterns necessary for accurate classification.

4.4 Discussion

The experimental results clearly demonstrate the effectiveness of the proposed SMART-NIR architecture in comparison to conventional transformer-based models. A key contributor to this performance gain is the incorporation of KAN as a classifier. As shown in Table 2, replacing the standard MLP with KAN resulted in a substantial increase in classification accuracy, approximately +5%, regardless of the kernel size. This confirms KAN's superior ability to capture non-linear relationships within spectral data, which is critical in NIR classification tasks.

Although prior studies [29] have indicated that KAN tends to show significant advantages primarily in symbolic formula representation tasks, this insight can help explain why KAN is particularly effective in the SMART-NIR setting. NIR spectral data, despite being real-valued, often reflects underlying physical or chemical relationships that are structured, continuous, and locally smooth, properties com-

Table 4: Comparison performance of model using MLP and Dual-MLP with KAN as classifier, multi-kernel and $N_L = 6$, $N_h = 6$ and $C_{out} = 64$ (noise adding).

Network	Size (params)	FLOPs (M)	Accuracy (%)
MLP	1211K	66.1	99.00 ± 0.34
Dual-MLP	**917K**	**47.2**	$\mathbf{99.01 \pm 0.36}$

Table 5: Performance comparison of SMART-NIR (Multi-kernel input embedding, Dual-MLP and KAN classifiers) and baseline Transformer (single kernel input embedding, standard MLP encoder and classifier), under varying hyperparameters ($C_{out} = 64$, noise adding).

N_L	N_h	Accuracy (%)	
		SMART-NIR	**Transformer**
4	4	$\mathbf{98.95 \pm 0.40}$	89.03 ± 0.92
4	6	$\mathbf{98.96 \pm 0.37}$	90.18 ± 0.75
6	6	$\mathbf{99.01 \pm 0.36}$	91.29 ± 0.52

monly found in symbolic or functional domains. Thus, the NIR classification task shares characteristics with symbolic function approximation, where KAN has proven to excel. Therefore, the performance gain observed in SMART-NIR can be interpreted as a natural extension of KAN's strength in modeling structured, low-dimensional, and function-like patterns. The use of learnable B-spline activation functions allows KAN to adaptively fit localized variations in spectral features, enabling more expressive representations than fixed non-linearities typically used in MLPs. This property is particularly beneficial for NIR data, which requires nuanced modeling of small absorption peaks or shifts.

Meanwhile, as reported in Table 5, SMART-NIR consistently outperforms the baseline Transformer across all examined configurations, with an accuracy gain of up to 8%, while also maintaining lower computational complexity due to architectural refinements such as the Dual-MLP and the integration of KAN. Furthermore, the introduction of Dual-MLP not only improved model accuracy but also significantly reduced computational costs. As detailed in Table 4, the Dual-MLP module achieved equivalent or better performance than its standard MLP counterpart while reducing both the parameter count and FLOPs by approximately 25%. This efficiency makes SMART-NIR especially suitable for deployment in resource-constrained environments, where model compactness and speed are critical.

While these results are promising, it is important to acknowledge the limitations related to the dataset size. The experiments were conducted on a relatively small dataset comprising only 1028 samples, which may raise concerns regarding model generalization. Although SMART-NIR demonstrated stable performance across 5-fold cross-validation, as shown in Table 6, further validation on larger and more diverse datasets is necessary to confirm the model's robustness and applicability to broader NIR sensing tasks.

Lastly, the ablation studies on kernel size and transformer hyperparameters (Tables 2 and 3) offer additional insights into the model's design. The multi-kernel configuration led to improved feature extraction, and optimal settings of $N_L = 6$, $N_h = 6$, and $C_{out} = 64$ were found to balance accuracy and complexity most effectively. These findings highlight the importance of careful architectural and hyperparameter selection in achieving state-of-the-art performance.

5 Conclusion

This study proposed SMART-NIR, a novel adaptive representation architecture for NIR spectral analysis, combining three main components: a multi-kernel block for multi-scale feature extraction, an improved Vision Transformer Encoder with Dual-MLP to capture global relationships, and KAN for nonlinear classification.

Our experimental results demonstrate the significant performance of SMART-NIR with an average accuracy of 99.24% in classifying four food categories through 5-fold cross-validation, outperforming the traditional Transformer by about 8%. The model balances performance and computational complexity, significantly reducing the number of parameters and FLOPs. Specifically, Dual-MLP reduces the number of parameters and FLOPs by 25% compared to the standard MLP while maintaining high accuracy. Multi-kernel block demonstrates efficient feature extraction at multiple scales. Notably, KAN improves the accuracy by about 5% compared to the traditional MLP, demonstrating the ability to approximate complex functions with few parameters accurately.

The optimal configuration was determined to be $N_L = 6$, $N_h = 6$, and $C_{out} = 64$, balancing performance and complexity. Cross-validation results showed the model has stable performance with high accuracy, precision, sensitivity, and F1-score across folds. These improvements effectively handle the complexity of NIR spectral data, balancing high accuracy and computational efficiency. This opens up the potential for broad application in near-infrared spectrum classification tasks, especially when integrated on mobile devices or processing spectra measured in real-world environments with high variability.

Building on this research, we intend to extend the identification and prediction of chemical substances on NIR spectral data in various types of food collected from natural environments. Besides, real-time monitoring of the dynamic spectrum or optimizing computational resources is also a potential research direction. This research also serves as a

Table 6: Evaluation of SMART-NIR with optimal hyperparameters through 5-fold cross validation (w/o noise adding)

Fold	F1-Score	Precision	Recall	Accuracy
Fold 1	0.993	0.997	0.989	99.53
Fold 2	0.986	0.986	0.986	99.37
Fold 3	0.995	0.996	0.994	98.93
Fold 4	0.991	0.993	0.990	98.90
Fold 5	0.995	0.993	0.996	99.47
Average	$\mathbf{0.992 \pm 0.004}$	$\mathbf{0.993 \pm 0.005}$	$\mathbf{0.991 \pm 0.004}$	$\mathbf{99.24 \pm 0.32}$

Table 7: Comparison of inference time for 100 samples of SMART-NIR (Multi-kernel input embedding, Dual-MLP and KAN classifiers), and baseline Transformer architecture (single kernel input embedding, standard MLP encoder and classifier), under varying hyperparameters. ($C_{out} = 64$)

N_L	N_h	Time (ms)	
		SMART-NIR	**Transformer**
4	4	$\mathbf{198.4 \pm 14.7}$	235.2 ± 12.9
4	6	$\mathbf{212.3 \pm 16.1}$	248.6 ± 14.1
6	6	$\mathbf{293.4 \pm 15.3}$	377.9 ± 15.2

Table 8: The evaluation results of SMART-NIR with different stride values

Stride	Accuracy	Precision	Recall	F1-Score
1	$\mathbf{99.01 \pm 0.36}$	$\mathbf{0.99 \pm 0.03}$	$\mathbf{0.99 \pm 0.03}$	$\mathbf{0.99 \pm 0.03}$
2	98.50 ± 0.40	0.98 ± 0.03	0.97 ± 0.03	0.97 ± 0.03
3	97.80 ± 0.50	0.96 ± 0.04	0.96 ± 0.03	0.96 ± 0.03

basis for implementing and developing integrated mobile applications for rapid inspection tasks based on NIR spectra.

It can be seen that the SMART-NIR architecture not only contributes significantly to the food industry in particular but also has the potential to expand to safety quality inspection in many fields, corresponding to its ability to learn and adapt flexibly to different types of data. This strongly promotes research and dramatically contributes to ensuring quality safety in many areas of our real life.

Acknowledgement

This study is funded and implemented for the project with number 24/HD-KHCN/2023. This work is supported by the People's Committee, Da Nang, and the University of Science and Technology, University of Danang.

References

[1] Yue Huang. Chemometric methods in analytical spectroscopy technology. In *Chemometric Methods in Analytical Spectroscopy Technology*, pages 1–29. Springer, 2022. doi:10.1007/978-981-19-1625-0.

[2] Y.-H. Yun. Modern spectral analysis techniques. In *Chemometric Methods in Analytical Spectroscopy Technology*, pages 31–87. Springer Nature Singapore, Singapore, 2022. doi:10.1007/978-981-19-1625-0_2.

[3] Yukihiro Ozaki, Christian Huck, Satoru Tsuchikawa, and Søren Balling Engelsen. *Near-infrared spectroscopy: theory, spectral analysis, instrumentation, and applications*. Springer Nature, 2020. doi:10.1007/978-981-15-8648-4.

[4] Hanieh Nobari Moghaddam, Zahra Tamiji, Mahsa Akbari Lakeh, Mohammad Reza Khoshayand, and Mannan Haji Mahmoodi. Multivariate analysis of food fraud: A review of nir based instruments in tandem with chemometrics. *Journal of Food Composition and Analysis*, 107:104343, 2022. doi:10.1016/j.jfca.2021.104343.

[5] Krzysztof B Beć, Justyna Grabska, and Christian W Huck. Miniaturized nir spectroscopy in food analysis and quality control: Promises, challenges, and perspectives. *Foods*, 11(10):1465, 2022. doi:10.3390/foods11101465.

[6] JP Cruz-Tirado, Matheus Silva dos Santos Vieira, Oscar Oswaldo Vásquez Correa, Daphne Ramos Delgado, José Manuel Angulo-Tisoc, Douglas Fernandes Barbin, and Raúl Siche. Detection of adulteration of alpaca (vicugna pacos) meat using a portable nir spectrometer and nir-hyperspectral imaging. *Journal of Food Composition and Analysis*, 126:105901, 2024. doi:10.1016/j.jfca.2023.105901.

[7] Deepoo Meena, Somsubhra Chakraborty, and Jayeeta Mitra. Geographical origin identification of red chili powder using nir spectroscopy combined with simca and machine learning algorithms. *Food Analytical Methods*, 17(7):1005–1023, 2024. doi:10.1007/s12161-024-02625-6.

[8] Rui Zhu, Xiaohong Wu, Bin Wu, and Jiaxing Gao. High-accuracy classification and origin traceability of peanut kernels based on near-infrared (nir) spectroscopy using adaboost-maximum uncertainty linear discriminant analysis. *Current Research in Food Science*, 8:100766, 2024. doi:10.1016/j.crfs.2024.100766.

[9] Libo Yuan, Xiangru Meng, Kehui Xin, Ying Ju, Yan Zhang, Chunling Yin, and Leqian Hu. A comparative study on classification of edible vegetable oils by infrared, near infrared and fluorescence spectroscopy combined with chemometrics. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 288:122120, 2023. doi:10.1016/j.saa.2022.122120.

[10] Cristina Quintelas, Cláudia Rodrigues, Clara Sousa, Eugénio C Ferreira, and António L Amaral. Cookie composition analysis by fourier transform near infrared spectroscopy coupled to chemometric analysis. *Food Chemistry*, 435:137607, 2024. doi:10.1016/j.foodchem.2023.137607.

[11] Qiyong Jiang, Min Zhang, Arun S Mujumdar, and Dayuan Wang. Non-destructive quality determination of frozen food using nir spectroscopy-based machine learning and predictive modelling. *Journal of food engineering*, 343:111374, 2023. doi:10.1016/j.jfoodeng.2022.111374.

[12] Songguang Zhao, Tianhui Jiao, Selorm Yao-Say Solomon Adade, Zhen Wang, Xiaoxiao Wu, Huanhuan Li, and Quansheng Chen. Based on vis-nir combined with ann for on-line detection of bacterial concentration during kombucha fermentation. *Food Bioscience*, 60:104346, 2024. doi:10.1016/j.fbio.2024.104346.

[13] Enguang Zuo, Lei Sun, Junyi Yan, Cheng Chen, Chen Chen, and Xiaoyi Lv. Rapidly detecting fennel origin of the near-infrared spectroscopy based on extreme learning machine. *Scientific reports*, 12(1):13593, 2022. doi:10.1038/s41598-022-17810-y.

[14] Wenwen Zhang, Liyanaarachchi Chamara Kasun, Qi Jie Wang, Yuanjin Zheng, and Zhiping Lin. A review of machine learning for near-infrared spectroscopy. *Sensors*, 22(24):9764, 2022. doi:10.3390/s22249764.

[15] Zhuowen Deng, Tao Wang, Yun Zheng, Wanli Zhang, and Yong-Huan Yun. Deep learning in food authenticity: Recent advances and future trends. *Trends in Food Science & Technology*, 144:104344, 2024. doi:10.1016/j.tifs.2024.104344.

[16] Puneet Mishra, Dário Passos, Federico Marini, Junli Xu, Jose M Amigo, Aoife A Gowen, Jeroen J Jansen, Alessandra Biancolillo, Jean Michel Roger, Douglas N Rutledge, et al. Deep learning for near-infrared spectral data modelling: Hypes and benefits. *TrAC Trends in Analytical Chemistry*, 157:116804, 2022. doi:10.1016/j.trac.2022.116804.

[17] Mingrui Zhao, Hao Cang, Huixin Chen, Chu Zhang, Tianying Yan, Yifan Zhang, Pan Gao, and Wei Xu. Determination of quality and maturity of processing tomatoes using near-infrared hyperspectral imaging with interpretable machine learning methods. *Lwt*, 183:114861, 2023. doi:10.1016/j.lwt.2023.114861.

[18] Wilson Castro, Monica Saavedra, Jorge Castro, Adriano Rogério Bruno Tech, Tony Chuquizuta, and Himer Avila-George. Using recurrent neural networks to identify broken-cold-chain fish fillet from spectral profiles. *Neural Computing and Applications*, 36(8):4377–4386, 2024. doi:10.1007/s00521-023-09311-4.

[19] Fujia Dong, Yongzhao Bi, Jie Hao, Sijia Liu, Weiguo Yi, Wenjie Yu, Yu Lv, Jiarui Cui, Hui Li, Jinhua Xian, et al. A new comprehensive quantitative index for the assessment of essential amino acid quality in beef using vis-nir hyperspectral imaging combined with lstm. *Food Chemistry*, 440:138040, 2024. doi:10.1016/j.foodchem.2023.138040.

[20] Samet Ozturk, Alexander Bowler, Ahmed Rady, and Nicholas J Watson. Near-infrared spectroscopy and machine learning for classification of food powders during a continuous process. *Journal of Food Engineering*, 341:111339, 2023. doi:10.1016/j.jfoodeng.2022.111339.

[21] Seeun Jo, Woosuk Sohng, Hyeseon Lee, and Hoeil Chung. Evaluation of an autoencoder as a feature extraction tool for near-infrared spectroscopic discriminant analysis. *Food Chemistry*, 331:127332, 2020. doi:10.1016/j.foodchem.2020.127332.

[22] Bo Yang, Cheng Chen, Fangfang Chen, Chen Chen, Jun Tang, Rui Gao, and Xiaoyi Lv. Identification of cumin and fennel from different regions based on generative adversarial networks and near infrared spectroscopy. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 260:119956, 2021. doi:10.1016/j.saa.2021.119956.

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[24] Xiu Jin, Jun Zhou, Yuan Rao, XiaoDan Zhang, Wu Zhang, WenJing Ba, Xiaohu Zhou, and Tong Zhang. An innovative approach for integrating two-dimensional conversion of vis-nir spectra with the swin transformer model to leverage deep learning for predicting soil properties. *Geoderma*, 436:116555, 2023. doi:10.1016/j.geoderma.2023.116555.

[25] Zenghui Wang, Jun Zhang, Xiaochu Zhang, Peng Chen, and Bing Wang. Transformer model for functional near-infrared spectroscopy classification. *IEEE Journal of Biomedical and Health Informatics*, 26(6):2559–2569, 2022. doi:10.1109/JBHI.2022.3140531.

[26] You Li, Hongwei Sun, Yurui Zheng, Qiquan Wei, Zhaoqing Chen, Jianyi Zhang, Hengnian Qi, Chu Zhang, and Fengnong Chen. Combined gramian angular difference field image coding and improved mobile vision transformer for determination of apple soluble solids content by vis-nir spectroscopy. *Journal of Food Composition and Analysis*, 131:106200, 2024. doi:10.1016/j.jfca.2024.106200.

[27] Lingxiang Liao, Jingqing Lu, Lutao Wang, Yongqing Zhang, Dongrui Gao, and Manqing Wang. Ct-net: an interpretable cnn-transformer fusion network for fnirs classification. *Medical & Biological Engineering & Computing*, 62(10):3233–3247, 2024. doi:10.1007/s11517-024-03138-4.

[28] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljacic, Thomas Y. Hou, and Max Tegmark. KAN: Kolmogorov–arnold networks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL: https://surl.li/oivymi.

[29] Runpeng Yu, Weihao Yu, and Xinchao Wang. Kan or mlp: A fairer comparison, 2024. URL: https://arxiv.org/abs/2407.16674.

[30] C. Dong, L. Zheng, and W. Chen. Kolmogorov-arnold networks (kan) for time series classification and robust analysis. In Q.Z. Sheng and et al., editors, *Advanced Data Mining and Applications. ADMA 2024*, volume 15390 of *Lecture Notes in Computer Science*, Singapore, 2025. Springer. doi:10.1007/978-981-96-0840-9_24.

[31] Bihui Yu, Zong Wang, and Lijun Fu. Timeskan: A generic backbone network for time series prediction based on kolmogorov-arnold networks. In *2024 10th International Conference on Computer and Communications (ICCC)*, pages 1573–1579, 2024. doi:10.1109/ICCC62609.2024.10941880.

[32] Z. Wang, A. Zainal, M.M. Siraj, et al. An intrusion detection model based on convolutional kolmogorov-arnold networks. *Scientific Reports*, 15:1917, 2025. doi:10.1038/s41598-024-85083-8.

[33] Seyd Teymoor Seydi, Zavareh Bozorgasl, and Hao Chen. Unveiling the power of wavelets: A wavelet-based kolmogorov-arnold network for hyperspectral image classification. *arXiv preprint arXiv:2406.07869*, 2024. URL: https://arxiv.org/abs/2406.07869.

[34] I. Latreche, S. Slatnia, O. Kazar, et al. A review on deep learning techniques for eeg-based driver drowsiness detection systems. *Informatica*, 48(3), 2024. doi:10.31449/inf.v48i3.5056.

[35] N. T. H. Phuong, H. Nguyen Van, X. Nguyen Thi Thanh, and P. Nguyen Ngoc. Advances in machine learning framework for near-infrared spectroscopy: A taxonomic review on food quality assessment. *IJCAI*, 49(11):121–148, Jan 2025. doi:10.31449/inf.v49i11.7482.

[36] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006. doi:10.1162/neco.2006.18.7.1527.

[37] Wartini Ng, Budiman Minasny, Maryam Montazerolghaem, Jose Padarian, Richard Ferguson, Scarlett Bailey, and Alex B McBratney. Convolutional neural network for simultaneous prediction of several soil properties using visible/near-infrared, mid-infrared, and their combined spectra. *Geoderma*, 352:251–267, 2019. doi:10.1016/j.geoderma.2019.06.016.

[38] Xiaoyi Chen, Qinqin Chai, Ni Lin, Xianghui Li, and Wu Wang. 1d convolutional neural network for the discrimination of aristolochic acids and their analogues based on near-infrared spectroscopy. *Analytical Methods*, 11(40):5118–5125, 2019. doi:10.1039/C9AY01531K.

[39] Puneet Mishra and Dário Passos. Multi-output 1-dimensional convolutional neural networks for simultaneous prediction of different traits of fruit based on near-infrared spectroscopy. *Postharvest Biology and Technology*, 183:111741, 2022. doi:10.1016/j.postharvbio.2021.111741.

[40] Jiechao Yang, Xuelei Wang, Ruihua Wang, and Huanjie Wang. Combination of convolutional neural networks and recurrent neural networks for predicting soil properties using vis–nir spectroscopy. *Geoderma*, 380:114616, 2020. doi:10.1016/j.geoderma.2020.114616.

[41] Van Hieu Nguyen, Le Huy Hien Ngo, Minh Toan Dinh, Binh Phan, Minh Nhat Phan, Thi Anh Phung, Viet Hung Le, and Huy Tuong Nguyen. NIRsViT: A novel deep learning model for manure identification using near-infrared spectroscopy and imbalanced data handling. *Cybernetics and Physics*, 13(4):323–333, 2024. doi:10.35470/2226-4116-2024-13-4-323-333.

[42] Jun Liu, Jianxing Zhang, Zhenglin Tan, Qin Hou, and Ruirui Liu. Detecting the content of the bright blue pigment in cream based on deep learning and near-infrared spectroscopy. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*,

270:120757, 2022. doi:10.1016/j.saa.2021.120757.

[43] Lingqiao Li, Xipeng Pan, Wenli Chen, Manman Wei, Yanchun Feng, Lihui Yin, Changqin Hu, and Huihua Yang. Multi-manufacturer drug identification based on near infrared spectroscopy and deep transfer learning. *Journal of Innovative Optical Health Sciences*, 13(04):2050016, 2020. doi:10.1142/S1793545820500169.

[44] Eui Jung Moon, Youngsik Kim, Yu Xu, Yeul Na, Amato J Giaccia, and Jae Hyung Lee. Evaluation of salmon, tuna, and beef freshness using a portable spectrometer. *Sensors*, 20(15):4299, 2020. doi:10.3390/s20154299.

[45] Wartini Ng, Budiman Minasny, Wanderson de Sousa Mendes, and José Alexandre Melo Demattê. The influence of training sample size on the accuracy of deep learning models for the prediction of soil properties with near-infrared spectroscopy data. *Soil*, 6(2):565–578, 2020. doi:10.5194/soil-6-565-2020.

[46] Puneet Mishra and Dário Passos. A synergistic use of chemometrics and deep learning improved the predictive performance of near-infrared spectroscopy models for dry matter prediction in mango fruit. *Chemometrics and Intelligent Laboratory Systems*, 212:104287, 2021. doi:10.1016/j.chemolab.2021.104287.

[47] Max Welling and Diederik P Kingma. Auto-encoding variational bayes. *CoRR*, 2014. URL: https://api.semanticscholar.org/CorpusID:216078090.

[48] Guang-Bin Huang, Zuo Bai, Liyanaarachchi Lekamalage Chamara Kasun, and Chi Man Vong. Local receptive fields based extreme learning machine. *IEEE Computational intelligence magazine*, 10(2):18–29, 2015. doi:10.1109/MCI.2015.2405316.

[49] Dong Xiao, Hongzong Li, and Xiaoyu Sun. Coal classification method based on improved local receptive field-based extreme learning machine algorithm and visible–infrared spectroscopy. *ACS omega*, 5(40):25772–25783, 2020. doi:10.1021/acsomega.0c03069.

[50] Dong Xiao, Quoc Huy Vu, and Ba Tuan Le. Salt content in saline-alkali soil detection using visible-near infrared spectroscopy and a 2d deep learning. *Microchemical Journal*, 165:106182, 2021. doi:10.1016/j.microc.2021.106182.

[51] Huazhou Chen, An Chen, Lili Xu, Hai Xie, Hanli Qiao, Qinyong Lin, and Ken Cai. A deep learning cnn architecture applied in smart near-infrared analysis of water pollution for agricultural irrigation resources. *Agricultural Water Management*, 240:106303, 2020. doi:10.1016/j.agwat.2020.106303.

[52] Eirik Almklov Magnussen, Johanne Heitmann Solheim, Uladzislau Blazhko, Valeria Tafintseva, Kristin Tøndel, Kristian Hovde Liland, Simona Dzurendova, Volha Shapaval, Christophe Sandt, Ferenc Borondics, et al. Deep convolutional neural network recovers pure absorbance spectra from highly scatter-distorted spectra of cells. *Journal of biophotonics*, 13(12):e202000204, 2020. doi:10.1002/jbio.202000204.

[53] Jingru Yang, Jin Wang, Guodong Lu, Shaomei Fei, Ting Yan, Cheng Zhang, Xiaohui Lu, Zhiyong Yu, Wencui Li, and Xiaolin Tang. Teanet: Deep learning on near-infrared spectroscopy (nir) data for the assurance of tea quality. *Computers and Electronics in Agriculture*, 190:106431, 2021. doi:10.1016/j.compag.2021.106431.

[54] Uladzislau Blazhko, Volha Shapaval, Vassili Kovalev, and Achim Kohler. Comparison of augmentation and pre-processing for deep learning and chemometric classification of infrared spectra. *Chemometrics and Intelligent Laboratory Systems*, 215:104367, 2021. doi:10.1016/j.chemolab.2021.104367.

[55] Ioannis Malounas, Wout Vierbergen, Sezer Kutluk, Manuela Zude-Sasse, Kai Yang, Ming Zhao, Dimitrios Argyropoulos, Jonathan Van Beek, Eva Ampe, and Spyros Fountas. Spectrofood dataset: A comprehensive fruit and vegetable hyperspectral metadataset for dry matter estimation. *Data in Brief*, 52:110040, 2024. doi:10.1016/j.dib.2024.110040.

[56] Dusan Agrez, Damir Ilic, and Janko Drnovsek. Signal and zero padding to improve parameters estimations of sinusoidal signals in the frequency domain. *Acta IMEKO*, 5(3):47–54, 2016. doi:10.21014/acta_imeko.v5i3.385.

[57] Luis B Almeida. Multilayer perceptrons. In *Handbook of Neural Computation*, pages C1–2. CRC Press, 2020. doi:10.1016/C2016-0-01217-2.

[58] Xinghao Chen, Gongyi Cheng, Shuhan Liu, Sizhuo Meng, Yiping Jiao, Wenjie Zhang, Jing Liang, Wang Zhang, Bin Wang, Xiaoxuan Xu, et al. Probing 1d convolutional neural network adapted to near-infrared spectroscopy for efficient classification of mixed fish. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 279:121350, 2022. doi:10.1016/j.saa.2022.121350.

[59] Penghui Sun, Jiajia Wang, and Zhilin Dong. Cnn–lstm neural network for identification of pre-cooked pasta products in different physical states using infrared spectroscopy. *Sensors*, 23(10):4815, 2023. doi:10.3390/s23104815.