Enhancing EEG-based Authentication with Transformer in Internet of Things

Chunxue Li, Weizhi Meng, *Senior Member, IEEE* and Wenjuan Li *Senior Member, IEEE*

***Abstract*—With the rapid growth of Internet of Things (IoT) and edge computing platforms, the Internet of Medical Things (IoMT) has become popular and important in healthcare industry, i.e., there is an increase of brainwave headsets and headbands. However, the security and privacy of shared data can be easily compromised if an attacker can access the IoMT devices and check all the data. There is a need to authenticate users before they can use the healthcare devices. For this reason, Electroencephalography (EEG) based authentication is a necessary security solution. In recent years, EEG-based authentication has witnessed significant advancements, but traditional models face challenges in capturing the complex spatial and temporal dependencies present in EEG signals. This work aims to address these limitations and explore the effect of Transformer model in the domain of EEG-based authentication. In particular, we devise a modified Vision Transformer model (ViT) to handle the specific characteristics of EEG data, such as spatial and temporal dependencies. In the evaluation, we compare our approach with the similar methods in the literature and examine the effect of fine-tune based on two datasets. The results demonstrate that our approach can effectively capture long-range dependencies and outperform conventional models.**

***Index Terms*—Data security, User authentication, Internet of Medical Things, Transformer model, Electroencephalography.**

I. Introduction

THE rapid growth of Internet of Things (IoT) enables the whole network to be fully distributed, where many Internet-enabled devices and components can be connected with each other [12]. Also, with the wide adoption of edge computing platforms, the IoT-Edge Continuum has become popular and important. Due to these new trends, healthcare data can be spread more distributed, since users would use various mobile / IoT devices to communicate with their doctors and medical organizations. Hence, Internet of medical Things (IoMT) [26] has become a new trend that requires timely and reliable delivery of the healthcare data in a distributed manner. Different from a traditional IoT device, Edge computing can provide IoMT with more resources such as memory, computational power, and network bandwidth [18], [35].

In recent years, Metaverse, as a future of digital connection, has provided a way of connecting physical and virtual space inspired via immersive technologies [21], e.g., virtual reality (VR), augmented reality and mixed reality. It is apparent that Metaverse devices will become common in the near future of IoMT, including brainwave headsets and headbands. That means more healthcare data would be transferred among Metaverse devices. However, if a cyber-attacker can access the healthcare devices, the security and privacy of patients' data can be easily compromised. Hence there is a need to protect the patients' data (e.g., diagnosis record) against unauthorized access on these devices. For instance, Zhu et al. [34] introduced SoundLock, a user authentication scheme using auditory-pupillary response as biometrics for VR devices. The corresponding pupillary response can be captured by the integrated eye tracker.

Currently, authentication systems are designed to verify the identity of users and grant them access to reasonable resources, such as online accounts, databases, or digital services, which play a critical role in ensuring the security and integrity of our personal and sensitive information. Traditionally, authentication methods have relied on something the user knows, such as passwords or PINs, as well as something they possess, such as physical tokens or smart cards. While in the era of Metavers, Electroencephalography (EEG) based authentication is a necessary and more available security solution, where users have to be authenticated based on their EEG signals [3]. EEG-based authentication has a substantial advantage due to the uniqueness of brainwave patterns. Each individual's neural connectivity and function are unique, resulting in highly individualized EEG signatures [33]. Unlike readily compromised passwords or physical tokens, brainwave patterns are inherently difficult to replicate or forge. Utilizing this distinction can add a layer of security and reduce the risk of unauthorized access to users' personal data such as multimedia data [17].

As presented in Figure 1, the human brain, which contains approximately 86 billion neurons, communicates primarily through electrical signals. Electroencephalography (EEG) is a non-invasive technique for measuring the electrical activity of the brain through the placement of electrodes on the scalp [32]. It detects and records the collective firing of neurons, which results in distinct brainwave patterns. These patterns are distinctive to each individual, making them possible biometric identifiers for authentication purposes [30].

Motivations. EEG-based authentication is a rapidly growing field that explores the use of EEG signals for building a secure and reliable user verification process, which is becoming more important in the era of IoMT. Due to the increasing technology advances and need for robust authentication methods, EEG-based authentication holds a great potential in providing a unique and efficient approach to address security concerns [24],

C. Li is with the Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark.

W. Meng is with the School of Computing and Communications, Lancaster University, United Kingdom.

E-mail: weizhi.meng@ieee.org (corresponding author)

W. Li is with Department of Mathematics and Information Technology, The Education University of Hong Kong, Hong Kong SAR.

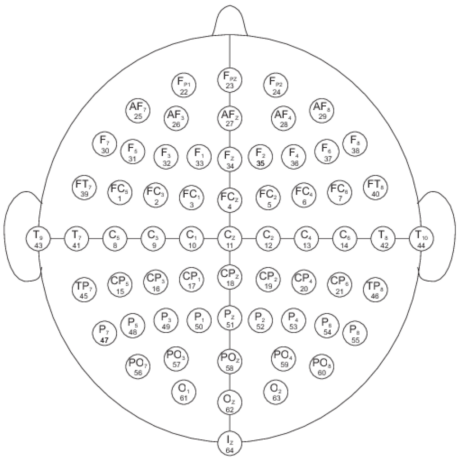


Fig. 1. 64-channel-sharbrough

[27], e.g., protecting the security and privacy of healthcare data stored on the devices. However, despite the advancements in this field, there are still significant challenges in the current literature.

One of the key challenges in existing EEG-based authentication is the development of accurate and reliable models that can effectively interpret and extract meaningful information from EEG signals [9]. Deep Learning approaches, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have been explored, but they often struggle to capture the complex spatial and temporal dependencies present in EEG signals [22]. This kind of limitation calls for the exploration of alternative models that can better handle the unique characteristics of EEG data.

We notice that the Transformer model, originally introduced for natural language processing tasks, has demonstrated a remarkable success in capturing long-range dependencies and modeling complex relationships within sequential data. Motivated by its performance, in this work, we aim to investigate the applicability of the Transformer model in the domain of EEG-based authentication and compare it with the existing state-of-the-art, especially CNN-based methods.

Contributions. By harnessing the power of self-attention and leveraging the ability to capture both local and global dependencies, Transformer model has the potential to overcome the limitations of existing EEG-based authentication approaches. For better performance, we design a modified Vision Transformer model (ViT) to handle the specific characteristics of EEG data, including spatial and temporal dependencies. Our contributions can be summarized as below.

- We investigate the applicability of Transformer model in the domain of EEG-based authentication, and develop an optimized Vision Transformer (ViT) model to make it more suitable for processing EEG data, e.g., handling spatial and temporal dependencies better.
- Then, we detail the preprocessing steps to enhance the quality of the input EEG data (such as noise removal, feature extraction, data normalization, and artifact removal.), and introduce the Optimized Vision Transformer (ViT) Model (e.g., self-attention mechanism).
- Furthermore, we conduct comprehensive experiments to examine the effect of fine-tune methods, evaluate the performance of our optimized ViT compared with the similar methods, and provide insights on the results.

Roadmap. The rest of this article is structured as follows: Section II presents related work on EEG-based authentication. Section III introduces the Transformer Model and our devised Vision Transformer (ViT) model for EEG-based authentication. Section IV describes the used datasets and preprocessing steps. Section V analyzes the evaluation results with discussion. Finally, we conclude this work in Section VI.

II. Related Work

EEG-based authentication refers to the verification of individuals using electroencephalogram (EEG) signals [33]. Due to its potential to provide secure and user-friendly authentication systems, this method has become an important security solution for Metaverse devices, which is the key to protect multimedia data stored on the devices.

Traditional machine learning approaches have been used for MI(Motor Imagery)-EEG signal classification, involving preprocessing, feature extraction, and classification. However, traditional methods face challenges such as noise interference, low signal-to-noise ratios (SNRs), subject dependency, and the need for manual feature extraction. These limitations have motivated the adoption of deep learning (DL) algorithms for MI-EEG classification. DL can automatically learn complex features from raw data, eliminating the need for extensive preprocessing and manual feature extraction. These DL models have shown remarkable performance in various fields and have been applied to EEG-based classification.

Table I shows the taxonomy of the deep learning models applied to EEG classification.

TABLE I
Taxonomy of Deep Learning Models for EEG Classification

Category	Examples of Models
Discriminative Models	- Convolutional Neural Networks (CNNs) - Recurrent Neural Networks (RNNs)
Representative Models	- Deep Autoencoders(DAE) - Deep Belief Networks (DBNs) - Restricted Boltzmann Machines (RBMs)
Generative Models	- Generative Adversarial Networks (GANs) - Variational Autoencoders (VAEs)
Hybrid Models	- CNN + LSTM - GNN + CNN

Discriminative Models, such as CNNs, were a popular DL model using EEG data. They excel in extracting local and spatial patterns from EEG data. CNN architectures consist of convolutional layers for feature extraction, pooling layers for dimensionality reduction, and fully connected (FC) layers for classification. Lawhern et al. [13] proposed a light CNN architecture with few parameters to classify different EEG tasks using raw signals and achieved remarkable performance. Olivas-Padilla et al. [16] then presented a CNN model for multiple MI classification with features extracted from MI data using the FBCSP approach. Xu et al. [23] provided a temporal-frequency image representation using a Wavelet Transform Time-Frequency coupled with a CNN model to classify MI signals, reaching an accuracy of 85.59% in a dataset with four MI classes.

Representative DL models refer to the DL architectures that specialize in feature extraction in an unsupervised manner, which can be used for various tasks, such as clustering and classification. Representative DL models include deep AEs (D-AEs), deep RBMs (D-RBMs), and DBN. Hassanpour et al. [10] proposed a stacked sparse AE model, defined as DBN-AE, for MI-EEG classification using FFT frequency features. Their study used a sliding window augmentation approach to increase the number of training data and achieved 71% accuracy using the public BCI-C IV-2a dataset.

Generative DL models, such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs), can be used for data augmentation to improve training performance. The GANs and VAEs proved effective in generating synthetic data that enhanced the classification accuracy of CNN models. Fahimi et al. [8] introduced a GAN-based generative model with light architecture for MI data argumentation, showing that increasing the number of training samples could improve the performance of the CNN model by 3.57% using the BCI-C III-4a dataset. The study also demonstrated the superiority of GAN models over VAEs.

Hybrid DL models, which combine different DL models into a single network, were employed in several studies. These models demonstrated promising results in MI classification. Examples include CNN combined with LSTM or GRU networks, CNN with Stacked Autoencoders (SAEs) or GANs, and DBNs combined with SVM classifiers. A hybrid CNN/RNN model, called recurrent convolutional neural network (RCNN), was proposed by the study in [19]. The model consisted of a single convolutional layer and four recurrent layers followed by a fully connected layer. The MI signal was converted to spectral images before being fed to the RCNN model. The performance of this model was studied using their local dataset consisting of two MI classes and three channels, reporting an accuracy of 77.72%.

Capturing Long-Range Dependencies: Transformer models excel at capturing long-range dependencies through the self-attention mechanism, allowing them to model complex patterns and understand contextual relationships more effectively. Du et al. [6] introduced an EEG Temporal-Spatial Transformer (ETST) model for accurate personal identification in cross-state scenarios by extracting information from EEG signals in the temporal and spatial domains. Hu et al. [11] presented AuthConformer and convolutional transformer model that can generate smartphone users' behavioral patterns for authentication, combining the capabilities of Transformers and convolutional layers to extract deep features from raw biometric data. Zeynali et al. [25] designed Transformer-based deep learning and ensemble models for EEG Classification that could outperform common deep learning models in EEG signal classification.

Research gap. To summarize, traditional learning methods encounter challenges such as noise interference, low signal-to-noise ratios, subject dependency, and the need for manual feature extraction. These limitations raise the demand of deep learning algorithms for EEG classification. In the literature, Transformer models have been explored for EEG classification and person identification, while to our knowledge, it has not been widely studied in EEG-based authentication. In this work, we aim to explore its performance for EEG-based authentication and develop an optimized ViT model. Our results show that Transformer models can enhance the authentication performance against unauthorized access to the device, e.g., protecting data security and privacy.

III. Our Proposed Approach

In this section, we introduce the background on Transformer model and its key architecture including Self-Attention, Positional Embedding, Decoder module. Then we detail our optimized Vision Transformer (ViT) Model for better handling the EEG data.

A. Background on Transformer Model

The Transformer model was a sequence-to-sequence model when it was first introduced, and the key feature is the use of self-attention [14]. To process a sequence, the most common thing that comes to mind is to use the Recurrent Neural Network (RNN). Its input is a sequence of vectors, the output is another sequence of vectors, and it can only compute sequentially from left to right or from right to left.

The model's ability to parallelize is constrained by this process, which makes the calculation at each time step dependent on the computation results of the earlier time steps. Also, the gradient disappearance problem will always exist with RNNs due to their inherent flaws, as shown in the following RNN Equation (1):

$$h(t) = f(W_{xh} \cdot x(t) + W_{hh} \cdot h(t-1) + b) \tag{1}$$

- `h(t)` represents the hidden state or output at time step t.
- `f()` is the activation function that introduces non-linearity to the hidden state.
- `x(t)` represents the input at time step t.
- W_{xh} is the weight matrix that connects the input to the hidden state.
- W_{hh} is the weight matrix that connects the hidden state to itself (recurrent connection).
- `b` is the bias vector.

The recursive nature of the equation comes from the term $W_{hh} \cdot h(t-1)$, which multiplies the previously hidden state `h(t-1)` by the recurrent weight matrix W_{hh}.

During the training process, the RNN learns to adjust the values of the weight matrices and the bias vector based on the provided input sequences and desired outputs. However, the issue of gradient disappearance or exploding gradients can arise when the weight matrix W_{hh} has a maximum eigenvalue greater than 1 or less than -1, respectively. In the case where the maximum eigenvalue of W_{hh} is less than 1, the gradients can become exponentially small, leading to difficulties in capturing long-term dependencies in the sequence data. Information loss occurs during sequential computation.

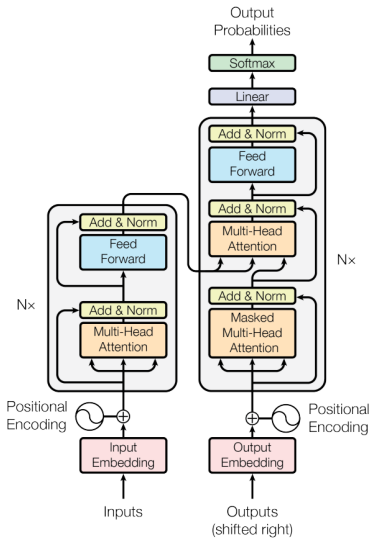


Fig. 2. The Transformer architecture

B. The Transformer Architecture

Figure 2 shows the Transformer architecture. The left part represents the encoder: the inputs consist of embeddings along with positional encodings. These inputs then pass through a uniform structure that can be repeated multiple times (N times), resulting in multiple layers (N layers). Each layer can be further divided into an attention layer and a fully connected layer, with additional processing steps such as skip connections and normalization layers.

The right part is similar to the decoder: the first input is the prefix information, followed by the embeddings generated in the previous step along with positional encodings. These inputs then go through a module that can be repeated multiple times. This module can be divided into three parts. The first part is an attention layer, the second part is cross-attention where the model attends to both the input sequence and an additional context sequence, and the third part is a fully connected layer. Skip connections and normalization layers are also utilized.

Finally, the output of the model is passed through a linear layer (fully connected layer) and then a softmax function generates the final predictions.

*a) **Self-Attention**:* First, we introduce three weight matrices: the query vector ($\mathbf{Q}$), key vector ($\mathbf{K}$), and value vector ($\mathbf{V}$). We perform matrix transformations as follows:

$$q_1 = x_1 \cdot \mathbf{W}^Q$$
$$k_1 = x_1 \cdot \mathbf{W}^K$$
$$v_1 = x_1 \cdot \mathbf{W}^V$$

$$q_2 = x_2 \cdot \mathbf{W}^Q$$
$$k_2 = x_2 \cdot \mathbf{W}^K$$
$$v_2 = x_2 \cdot \mathbf{W}^V$$

Through this process, different x_1 and x_2 can share the same weight matrices $\mathbf{W}^Q$, $\mathbf{W}^K$, and $\mathbf{W}^V$, allowing the exchange of information between them.

Next, we calculate z_1 and z_2 as follows:

$$z_1 = \theta_{11} \cdot v_1 + \theta_{12} \cdot v_2$$
$$z_2 = \theta_{21} \cdot v_1 + \theta_{22} \cdot v_2$$

To obtain the combination weights θ_{11} and θ_{12}, we use the softmax function on the attention scores:

$$[\theta_{11}, \theta_{12}] = \text{softmax}\left(\frac{q_1 k_1^T}{\sqrt{d_k}}, \frac{q_1 k_2^T}{\sqrt{d_k}}\right)$$

Similarly, for z_2, we calculate the combination weights θ_{21} and θ_{22}:

$$[\theta_{21}, \theta_{22}] = \text{softmax}\left(\frac{q_2 k_1^T}{\sqrt{d_k}}, \frac{q_2 k_2^T}{\sqrt{d_k}}\right)$$

In summary, the attention mechanism is defined as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \qquad (2)$$

In the given equations, d_k represents the dimensionality of the query or key vectors ($\mathbf{Q}$ or $\mathbf{K}$). Both of these vectors have the same dimensionality since they are used for dot product calculations. However, the value vectors ($\mathbf{V}$) may have a different dimensionality than the query and key vectors.

The division by $\sqrt{d_k}$ in the equations serves a purpose. It helps prevent the values of QK^T from becoming too large, especially when the dimensionality is high. This normalization is applied to avoid issues such as gradient vanishing during the backpropagation process.

Choosing $\sqrt{d_k}$ instead of d_k is an empirical choice. The purpose is to increase the value of QK^T to a reasonable extent without excessively inflating it. If we were to use d_k directly, it could potentially hinder the increase in the values of QK^T.

*b) **Multi-headed Attention**:* If we aim to use different W^Q, W^K, W^V, we can have different $\mathbf{Q}, \mathbf{K}, \mathbf{V}$. The multi-headed Attention mechanism employs multiple sets of W^Q, W^K, W^V, which allows for more diverse information. Each set of matrices provides a different perspective on the attention process and enhances the model's ability to attend to different aspects of the input sequence.

As shown in the below equation, the Multi-Head Attention can be obtained by concatenating the outputs of the individual attention heads:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\mathbf{W}_O$$
$$where \text{head}_i = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}).$$

*c) **Positional Embedding (PE)**:* When inputting data into the Transformer model, in addition to word vectors x, positional encoding is also added. The entire input embedding is obtained by adding the word embedding and positional embedding together. This allows the network to understand the position of each word in the input sentence. When performing self-attention, the network not only needs to know which word to focus on but also the relative distances between words.

Why is it important to know the relative positions of words? The Transformer model does not rely on recurrent neural networks (RNNs) or convolutional layers, so in order to utilize the sequential order of the input sequence, the model must be provided with positional information. Therefore, positional encoding is added at the bottom of the Encoder and Decoder modules. These positional encodings have the same dimensions as the input vectors, allowing them to be directly added together and inject positional information into the model.

The positional encoding equation used in the Transformer model is typically defined as follows:

$$\mathrm{PE}(pos, 2i) = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right)$$

$$\mathrm{PE}(pos, 2i+1) = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right)$$

where:

- pos represents the position of the word in the sequence.
- i corresponds to the dimension of the positional encoding vector.
- d_{model} represents the dimensionality of the model.

Here, the positional encoding values are calculated based on the sine and cosine functions, with varying frequencies determined by the position and the dimension. The encoding vectors are added element-wise to the word embedding vectors, providing the model with information about the relative positions of the words in the sequence.

It is worth noting that the above equations represent one common form of positional encoding, but there can be variations and alternative formulations based on specific requirements and research studies.

*d) **Decoder module**:* The Decoder module is structured similarly to the Encoder module. For the original Transformer model [20], the Decoder consists of N=6 stacked layers. Each layer is divided into three sub-layers. However, there are three main differences between the Encoder and the Decoder:

- Decoder SubLayer-1: This sub-layer utilizes a "Masked" Multi-Headed Attention mechanism to prevent the model from seeing future positions during the training. This masking helps prevent information leakage and ensures accurate predictions.
- SubLayer-2: This sub-layer is an Encoder-Decoder Multi-Head Attention mechanism. It allows the Decoder to attend to the input sequence provided by the Encoder, capturing the necessary context for generating accurate outputs.
- SubLayer-3: The output of SubLayer-3 is passed through Linear and Softmax layers to predict the probabilities of the corresponding words.

C. Our Optimized Vision Transformer (ViT) Model

The original Vision Transformer (ViT) model [7] is initially designed for image classification tasks. The key idea behind ViT is to represent an image as a sequence of fixed-size non-overlapping patches, and each patch is considered as a token similar to words in natural language processing. These patches are then linearly embedded into a high-dimensional vector space, forming the input to the Transformer network.

The Transformer's self-attention mechanism allows the model to attend to all the patches in the sequence simultaneously, capturing long-range dependencies and building a holistic representation of the image. The model then goes through several layers of self-attention and feed-forward neural networks, learning to recognize patterns and features at different levels of abstraction.

To further enhance the model's performance, Dosovitskiy et al. [7] introduced a pretraining stage, where the ViT has to be pretrained on a large dataset with a large number of image patches and a language modeling objective. After this pretraining phase, the model is fine-tuned on downstream computer vision tasks, such as image classification, object detection, and segmentation.

In the literature, the Vision Transformer has demonstrated impressive results, outperforming traditional CNN architectures on various computer vision benchmarks. One of its advantages is its ability to handle images of varying resolutions during training and inference, making it more adaptable to different input sizes. Here are some reasons why we might consider using ViT for handling EEG data:

- **Attention Mechanism:** ViT utilizes the self-attention mechanism, which allows the model to capture dependencies between different parts of the input data. This can be beneficial for EEG data, as it helps capture relationships between different EEG channels and time points.
- **Learn Global Patterns:** EEG data often contains global patterns and relationships that are essential for classification. ViT's self-attention mechanism enables the model to learn these global patterns effectively, as it considers all input positions simultaneously.
- **Hierarchical Representation:** ViT uses a hierarchical representation of the input data by splitting it into fixed-size patches. This approach can be advantageous for EEG data, as it helps capture both local and global patterns. By representing the EEG signals as patches, the model can learn spatial relationships between different brain regions.
- **Flexible Architecture:** ViT is a flexible architecture that can handle variable-sized input data. This is beneficial for EEG data, as EEG signals can have different lengths and numbers of channels. The model can also help process variable-length EEG signals by dividing them into fixed-size windows or segments.
- **Transfer Learning:** ViT has achieved state-of-the-art performance on various image classification tasks. By leveraging pre-trained ViT models trained on large-scale image datasets, it can benefit from transfer learning. Fine-tuning a pre-trained ViT model on the EEG data can help improve classification performance, especially when the labeled EEG data is limited.

ViT takes an image as input and divides it into fixed-size non-overlapping patches. These patches are then flattened and linearly embedded into token representations, which are fed into the transformer encoder. However, the input data for the EEG signal processing method consists of EEG signals, which are typically time-series data recorded from multiple electrode

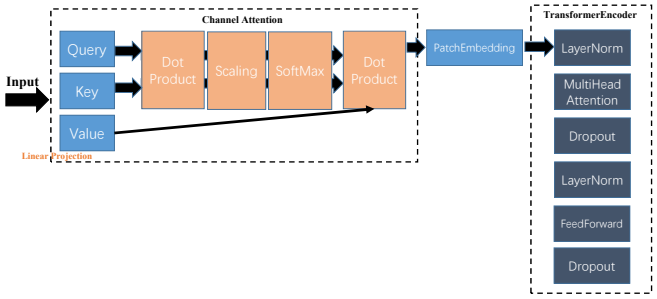


Fig. 3. The modified ViT model architecture

locations on the scalp. Thus, there is a need to optimize the ViT for better performance under EEG-based authentication.

A novel method for efficiently processing EEG signals should consider the importance of different feature channels and avoiding interference between them. Inspired by the scaled dot-product attention [20], a feature channel weighting technique was proposed. This method learns the dependence of each element on others to calculate the importance score for weighting data values. Initially, the input data is linearly transformed into vectors, including queries (Q) and keys (K) with dimension dk, and values (V) with dimension dv, along the spatial feature dimension.

Optimized ViT. In this work, we thus optimize the ViT by using the dot product to evaluate the correlation between one feature channel and all others, as shown in Figure 3– Channel Attention part. The scaled dot-product attention is applied to the dot-product result, dividing it by a scaling factor of $[\sqrt{dk}]$ to enhance the Softmax function's perception ability. The output weight score is assigned to V, generating the final representation using dot product. This process can be mathematically expressed as:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Where Attention(Q,K,V) represents the weighted representation, and Q, K, and V are matrices packed by vectors for simultaneous calculation.

As brain-driven behaviors are complete processes, we aim to effectively utilize the relationship between any two parts of a trial. To reduce computational complexity, we initially compress the data to one dimension. Since feature channels are already weighted in the previous step, we divide the data into multiple slices for attention training. For temporal transforming, we employ multi-head attention (MHA) to allow the model to learn dependencies from different angles. The input is split into h smaller parts called "heads," which perform attention in parallel. The outputs of each part are concatenated and linearly transformed to obtain the original size. As depicted in Figure 3 (TransformerEncoder part), This process can be represented as follows:

$$\text{MHA}(XQ, XK, XV) = [\text{head}_0; \ldots; \text{head}_{h-1}]W_o$$

$$\text{head}_i = \text{Attention}(XQW_{Qi}, XKW_{Ki}, XVW_{Vi})$$

Where $[.;.]$ denotes a concatenation operation, $XQ \in \mathbb{R}^{d\times d_k}$,$XK \in \mathbb{R}^{d\times d_k}$, and $XV \in \mathbb{R}^{d\times d_v}$ are linear transformations to obtain queries, keys, and values matrices of each head. $W_{Qi} \in \mathbb{R}^{d\times d_k}$, $W_{Ki} \in \mathbb{R}^{d\times d_k}$, and $W_{Vi} \in \mathbb{R}^{d\times d_v}$ are linear transformations to obtain queries, keys, and values matrices of each head. $W_o \in \mathbb{R}^{d_v\times d}$ is the linear transformation to obtain the final output. Additionally, a feed-forward (FF) block, consisting of two fully-connected layers with the GeLU activation function, is connected behind the MHA to enhance the model's perception and non-linear learning capabilities.

Layer normalization is applied before the MHA and FF block, and residual connections are used for better training. The module with MHA and FF block can be repeated for an ensemble effect. However, the temporal transforming method may capture dependencies between different slices but may overlook the position information, which is the sequence relationship between EEG sample points. To address this, we employ a convolutional layer on the time dimension to encode position information (PatchEmbedding) before compressing and slicing.

The final stage is the classification. A global pooling operation is applied to average all slices in the transforming part. The pooling result is then connected to a fully-connected layer after layer normalization. The number of output neurons is equal to the number of categories, and the Softmax function is used to obtain the predicted probability. The objective function is the classification loss achieved by cross-entropy, which can be represented as:

$$L = -\frac{1}{M}\sum_{n=1}^{N}\sum_{m=1}^{M} y_{nm}\log(\hat{y}_{nm})$$

Where M is the number of trials, N is the number of categories, y_{nm} denotes the real label for the $m-th$ trial, and $\hat{y}_{nm}$ represents the predicted probability of the $m-th$ trial for category n.

IV. EEG Data Preprocessing

In this section, we first describe the EEG datasets used in our evaluation: the PhysioNet EEG Motor Movement/Imagery Dataset (a large dataset) and the BCI Competition IV Dataset 2a (a small dataset). We then outline the preprocessing steps made to ensure data quality and explain the train-test split strategy.

A. EEG Dataset Description

A summary of the two EEG datasets is presented in Table II.

1) The PhysioNet Motor Movement/Imagery Dataset: This data set consists of over 1500 one- and two-minute EEG recordings, obtained from 109 volunteers.

Subjects performed different motor/imagery tasks while 64-channel EEG were recorded using the BCI2000 system (http://www.bci2000.org). Each subject performed 14 experimental runs: two one-minute baseline runs (one with eyes open, one with eyes closed), and three two-minute runs of each of the four following tasks:

TABLE II
EEG DATASET SUMMARY

Name	Subjects	Channels	Classes	Trials / class	Trials length	Sampling rate
PhysionetMI	109	64	4	23	4s	160Hz
BCI IV 2a	9	22	4	144	3s	100HZ

- A target appears on either the left or the right side of the screen. The subject opens and closes the corresponding fist until the target disappears. Then the subject relaxes.
- A target appears on either the left or the right side of the screen. The subject imagines opening and closing the corresponding fist until the target disappears. Then the subject relaxes.
- A target appears on either the top or the bottom of the screen. The subject opens and closes either both fists (if the target is on the top) or both feet (if the target is on the bottom) until the target disappears. Then the subject relaxes.
- A target appears on either the top or the bottom of the screen. The subject imagines opening and closing either both fists (if the target is on the top) or both feet (if the target is on the bottom) until the target disappears. Then the subject relaxes.

The EEG signals were recorded from 64 electrodes as per the international 10-10 system (excluding electrodes Nz, F9, F10, FT9, FT10, A1, A2, TP9, TP10, P9, and P10)

2) The BCI Competition IV Dataset 2a: The EEG Motor Movement/Imagery Dataset [1] includes EEG data from 9 people and focuses especially on 4-class motor imagery tasks. It is a commonly used benchmark dataset in the field of motor imagery categorization tasks using brain-computer interfaces. The cue-based BCI paradigm included four different motor imagery tasks: imagining the movement of the tongue (class 4), both feet (class 3), the left hand (class 1), and the right hand (class 2). For each subject, two sessions on various days were recorded. There are six runs in each session, separated by brief rest periods. There are 288 trials in each session, or 48 trials in each run (12 for each of the four potential classes).

A recording of about 5 minutes was made at the start of each session to gauge the EOG (Electrooculography) influence. Three segments were taken from the recording: two minutes of open eyes (gazing at a fixation cross on the screen), one minute of closed eyes, and one minute of eye movements.

The subjects were sitting in a comfortable armchair in front of a computer screen. At the beginning of a trial (t = 0 s), a fixation cross appeared on the black screen. In addition, a short acoustic warning tone was presented. After two seconds (t = 2 s), a cue in the form of an arrow pointing either to the left, right, down, or up (corresponding to one of the four classes left hand, right hand, foot, or tongue) appeared and stayed on the screen for 1.25 seconds. This prompted the subjects to perform the desired motor imagery task. No feedback was provided. The subjects were asked to carry out the motor imagery task until the fixation cross disappeared from the screen at t = 6 s. A short break followed when the screen was black again.

B. Data Preprocessing

Data preprocessing is an essential step in working with EEG data to improve the quality and reliability of the signals. In the context of EEG data, preprocessing involves converting raw data into a format that is more suitable for analysis and interpretation.

1) Filter: In EEG data processing, filtering refers to the application of digital filters to the raw EEG signal in order to remove unwanted noise or extract specific frequency bands of interest. Filters are used to shape the frequency content of the EEG signal and improve the signal-to-noise ratio.

There are different types of filters commonly used in EEG data processing, including high-pass filters, low-pass filters, band-pass filters, and notch filters:

- *High-pass filter:* A high-pass filter attenuates or removes low-frequency components from the EEG signal, allowing only high-frequency components to pass through. This filter is useful for removing drift, baseline wander, and other low-frequency artifacts.
- *Low-pass filter:* A low-pass filter attenuates or removes high-frequency components from the EEG signal, allowing only low-frequency components to pass through. This filter is used to remove high-frequency noise, such as electromagnetic interference.
- *Band-pass filter:* A band-pass filter allows a specific frequency range, known as the passband, to pass through while attenuating frequencies outside this range. It is useful for isolating specific frequency bands of interest, such as alpha (8-12 Hz) or beta (12-30 Hz) rhythms.
- *Notch filter:* A notch filter is designed to attenuate a narrow range of frequencies, typically centered around the powerline frequency (e.g., 50 Hz or 60 Hz) and its harmonics. It is used to remove powerline noise and related interference.

To implement filtering in EEG data processing, digital filter designs are typically used, such as Butterworth, Chebyshev, or elliptic filters. These filters can be implemented using various algorithms, such as finite impulse response (FIR) or infinite impulse response (IIR) filters. The filter parameters, such as the cutoff frequency or the width of the passband or stopband, can be adjusted based on the specific requirements of the EEG analysis.

The tuning of filter parameters was guided by a mix of established knowledge and experimental analysis. Refining Through Experimentation: Once the initial parameters were set, they were further adjusted through an iterative process. This involved: a) Testing different configurations and evaluating how they affected the quality of the signals, specifically their clarity and the preservation of meaningful patterns, and b) Assessing the model's performance during validation to identify the settings that could help retain features critical for

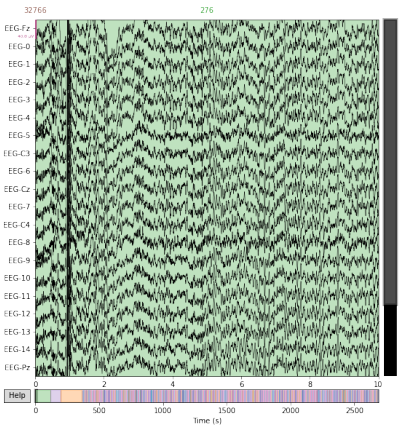


Fig. 4. Plot of raw EEG data.

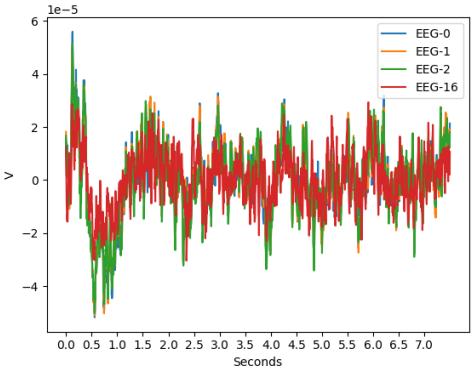


Fig. 5. Comparison of EEG-0, EEG-1, EEG-2, and EEG-16 channels with left hand event.

classification, while effectively minimizing noise and irrelevant information.

2) Artifact: Regarding EEG data processing, artifacts refer to unwanted signals or noise that can contaminate the recorded EEG signal. These artifacts can arise from various sources, including physiological sources (e.g., eye blinks, muscle activity), environmental sources (e.g., electrical interference), or technical sources (e.g., electrode movement or malfunction). Artifacts can significantly affect the quality and interpretability of EEG data, and thus, it is important to identify and mitigate them.

C. Data Analysis and Exploration

Data analysis and exploration are also crucial steps in understanding the EEG data and identifying potential artifacts. Visualization techniques can be used to examine the presence of artifacts and their impact on the data. Techniques such as plotting EEG channels, comparing shared variance between different electrodes, and analyzing artifact-related patterns can provide insights into the data quality and help select appropriate preprocessing techniques.

First, the raw EEG data is plotted from participant A01 in the BCI-IV 2a dataset, as shown in Figure 4.

Next, a single trial is isolated for further analysis. The events from the annotations are extracted, and the events of interest (left hand, right hand, foot, and tongue imagery trials) are identified. EEG-0, EEG-1, EEG-2, and EEG-16 channels with

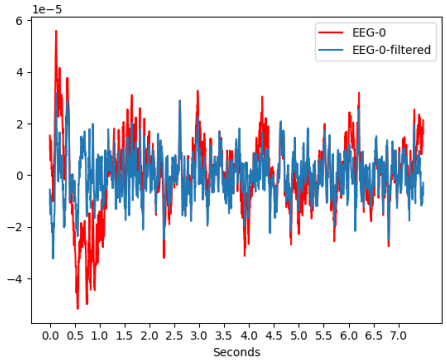


Fig. 6. Application of Butterworth filter to remove EEG noise.

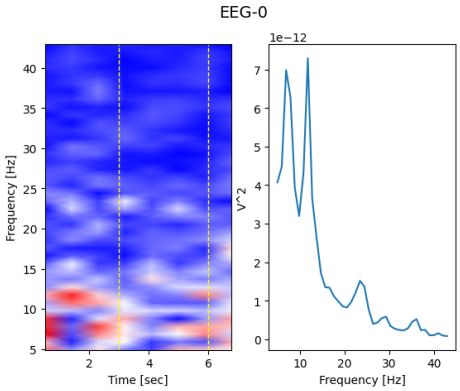


Fig. 7. Event-related desynchronization (ERD) analysis

left hand event are plotted in Figure 5, revealing artifacts related to blinks and eye movements.

From the plot, we can observe two interesting things:

- From seconds 0 to 0.5, we see a sharp oscillation where both channels are positively correlated. This pattern resembles a blink artifact, although the amplitude of the oscillation is relatively small.
- From seconds 0.5 to 1.5, there is a low-frequency signal oscillation where both EEG channels are highly negatively correlated. This pattern suggests a slight sideways or up-down eye movement.

To mitigate EOG artifacts, a high-pass, low-frequency filter is designed using a Butterworth filter. The filtered EEG channel (EEG-0) is compared to the unfiltered channel in Figure 6, demonstrated the effectiveness of the filtering process in reducing noise and removing artifacts.

Event-related desynchronization (ERD) analysis was also performed to investigate changes in oscillatory activity during motor imagery. The power spectrum and spectrogram of the EEG-0 channel are calculated and plotted in Figure 7.

The left part of Figure 7 is the Signal's Power Spectrum, which can give us the signal power across the frequency domain. The right part of Figure 7 is Spectrogram, which can give us the signal power across frequencies and also across time. Thus, we can have an idea of when signal changes in the frequency magnitude domain start to happen within each trial. This analysis also confirms that most changes in the frequency domain occur between 7-35Hz, which is consistent with sensorimotor tasks. Strong desynchronization is observed

in the 8-9Hz band during motor imagination, indicating a characteristic ERD response.

Algorithm 1: Data Split Strategy

Input: Dataset $D = (X, Y)$, validation split $\alpha_{\text{val}} = 0.10$, test split $\alpha_{\text{test}} = 0.20$
Output: $(X_{\text{train}}, Y_{\text{train}})$, $(X_{\text{val}}, Y_{\text{val}})$, $(X_{\text{test}}, Y_{\text{test}})$
Initialize empty arrays: $X_{\text{train}}, Y_{\text{train}}, X_{\text{val}}, Y_{\text{val}}, X_{\text{test}}, Y_{\text{test}}$
Compute the total number of classes:
$n_{\text{classes}} \leftarrow$ number of distinct labels in Y
for *each class* c *in* $n_{classes}$ **do**
end
Retrieve indices I_c of all samples belonging to class c
Shuffle I_c to randomize sample order
Compute number of samples for validation and testing:

$$n_{\text{val}} \leftarrow \lfloor \alpha_{\text{val}} \times |I_c| \rfloor, \quad n_{\text{test}} \leftarrow \lfloor \alpha_{\text{test}} \times |I_c| \rfloor$$

Select indices for validation and testing:

$$\text{val\_idx} \leftarrow \text{first } n_{\text{val}} \text{ indices from } I_c$$

$$\text{test\_idx} \leftarrow \text{next } n_{\text{test}} \text{ indices from } I_c$$

$$\text{train\_idx} \leftarrow \text{remaining indices in } I_c$$

Append samples to corresponding sets:

$$X_{\text{val}}, Y_{\text{val}} \leftarrow \text{append samples and labels from val\_idx}$$

$$X_{\text{test}}, Y_{\text{test}} \leftarrow \text{append samples and labels from test\_idx}$$

$$X_{\text{train}}, Y_{\text{train}} \leftarrow \text{append samples and labels from train\_idx}$$

Shuffle $X_{\text{train}}, Y_{\text{train}}$; $X_{\text{val}}, Y_{\text{val}}$; $X_{\text{test}}, Y_{\text{test}}$ to introduce randomness
Return: $(X_{\text{train}}, Y_{\text{train}})$, $(X_{\text{val}}, Y_{\text{val}})$, $(X_{\text{test}}, Y_{\text{test}})$

D. Data Split Strategy

The `split_data` function aims to partition the dataset into distinct subsets for training, validation, and testing purposes. It follows a systematic approach to ensure representative and unbiased subsets. Regarding the classification process, the current approach uses a fixed data split rather than repeated K-fold cross-validation. While this provides a baseline for performance, K-fold validation was not applied in this study. Incorporating K-fold validation in future work is acknowledged as a step to enhance the robustness of the results.

Our data split strategy is summarized in **Algorithm 1**, which ensures that each class contributes to the training, validation, and testing subsets in proportion to their representation in the original dataset. By randomly selecting samples for validation and testing without replacement, the function guarantees the absence of duplicate samples across the subsets. The shuffling step further promotes unbiased model evaluation and enhances generalization capabilities on unseen data. Normally, the percentage of train data, validation data, and test data in our code are 70%, 10%, and 20%.[1]

V. Evaluation and Analysis

In this section, we aim to evaluate the performance of our proposed ViT model as compared with the original Transformer model, CNN and the similar methods.

A. Experiment I: The Original Transformer Model

The BCI Competition IV Dataset 2a was chosen as the dataset, which is a commonly used EEG dataset in brain-computer interface research. The EEG data consisted of 22 channels or electrodes. The model was trained for 100 epochs with a learning rate of 0.0002. Preprocessing techniques were applied to the data, including a band-pass filter to remove unwanted frequency components and a Common Spatial Patterns (CSP) algorithm.

The parameters used for the first working solution are shown in Table III.

TABLE III
THE PARAMETERS OF V1

PARAMETER	TYPE	VALUE
dataset	null	BCI Competition IV Dataset 2a
channel	int	22
epoch	int	100
learining-rate	float	0.0002
preprocess	bool	True

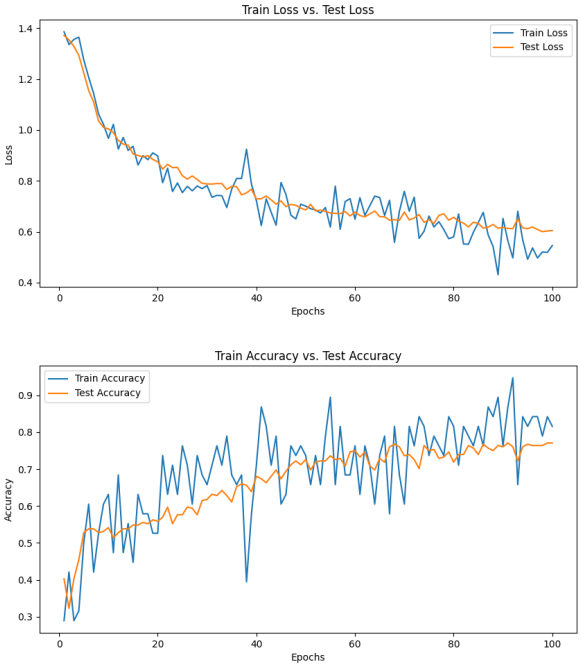


Fig. 8. The performance of the original Transformer model

*a) **Testing Analysis**:* From Figure 8, the train accuracy values range from approximately 0.29 to 0.95. The accuracy increases over the training epochs, indicating that the model is effectively learning and fitting the training data. Initially, the accuracy is relatively low (around 0.29) and gradually

[1]Our code is publicly available at: https://github.com/snow1/transformer

improves over time. The final train accuracy achieved is approximately 0.95, indicating that the model can accurately predict the classes for the training data.

The test accuracy values range from approximately 0.40 to 0.77. The accuracy fluctuates throughout the training process, indicating that the model's performance on unseen data varies across different epochs. The average accuracy is 0.696, and the best accuracy is 0.778. However, the accuracy values seem to reach a plateau of around 0.75, indicating that the model's performance may not significantly improve beyond that point.

The train loss values range from approximately 0.43 to 1.38. The loss decreases over the training epochs, indicating that the model is effectively optimizing the objective function and learning from the training data. Initially, the loss is relatively high (around 1.38) and gradually decreases over time. The final train loss achieved is approximately 0.55, indicating that the model is able to fit the training data relatively well.

The test loss values range from approximately 0.61 to 1.37. The loss fluctuates throughout the training process, suggesting that the model's generalization performance may vary across different epochs. The lowest test loss achieved is around 0.61, indicating that the model is able to achieve good performance on the test set. However, the loss values seem to stabilize around 0.65, suggesting that the model may struggle to improve its performance beyond that point.

Based on the obtained data, it seems that the model's performance is relatively stable, achieving moderate to good accuracy and relatively low loss values. However, the model's performance might not significantly improve beyond a certain point, indicating the possibility of reaching a performance plateau. It is worth mentioning that we used a band-pass filter to remove unwanted frequency components. Frequencies below 1 Hz were attenuated to eliminate slow drifts and baseline fluctuations, while frequencies above 50 Hz were suppressed to reduce high-frequency noise, including muscle artifacts and environmental interference. Following by this, CSP was also applied in the preprocessing but the result was similar without the preprocessing.

B. Experiment II - Our Proposed ViT

The Modified ViT model utilized specific parameter settings to customize its architecture and training process. We also selected the BCI Competition IV Dataset 2a, consisting of EEG data recorded from 22 channels. The model was trained for 100 epochs using a learning rate of 0.0002. Preprocessing techniques were not applied to the data in this case, as indicated by the "preprocess" parameter set to False.

It is worth noting that we selected a small value of 0.0002 as it is a commonly used and empirically effective learning rate in AI model training. Especially for complex models like Transformers and datasets with smaller size or higher noise levels (such as EEG data), this learning rate helps ensure stable convergence and prevents excessive parameter updates that could lead to training divergence.

Two important parameters, namely depth and emb_size, were specifically chosen to configure the Modified ViT model. The depth parameter represents the number of transformer encoder layers in the model. Each transformer encoder layer comprises self-attention and feed-forward sub-layers, enabling the model to capture intricate patterns and dependencies within the EEG data. Increasing the depth allows the model to handle more complex relationships but also escalates the computational complexity and memory requirements.

On the other hand, emb_size denotes the embedding size or the dimensionality of the patch embeddings in the model. The input EEG signals are partitioned into fixed-size patches, which are then linearly projected into a lower-dimensional space known as patch embeddings. These patch embeddings are subsequently fed into the transformer encoder layers for further processing. The emb_size parameter determines the dimensionality of these patch embeddings. Choosing a larger emb_size can enable the model to capture more detailed information and fine-grained features. However, it is essential to note that increasing the value of emb_size also leads to higher computational demands for the model.

By effectively selecting and configuring these parameters, our ViT model can adapt to the specific characteristics and complexities of EEG datasets. The chosen depth allows for capturing intricate patterns and dependencies, while the value of emb_size determines the level of detail and granularity that can be represented in the model. This parameter configuration facilitates the learning and representation of EEG data, enhancing the model's ability to analyze and classify brain activity accurately.

The parameters used for this experiment on our optimized ViT are shown in Table IV.

TABLE IV
THE PARAMETERS OF MODIFIED VIT

PARAMETER	TYPE	VALUE
dataset	null	BCI Competition IV Dataset 2a
channel	int	22
epoch	int	100
learining-rate	float	0.0002
preprocess	bool	False
depth	int	2
emb_size	int	5

In the table, depth refers to the number of transformer encoder layers in the model. Each transformer encoder layer consists of multiple self-attention and feed-forward sub-layers. Increasing the depth allows the model to capture more complex patterns and dependencies within the image data. However, increasing depth also increases the computational complexity and memory requirements of the model.

In particular, emb_size represents the embedding size or the dimensionality of the patch embeddings in the model. In the model, the input image is divided into fixed-size patches, and each patch is linearly projected to a lower-dimensional space. The resulting patch embeddings are then fed into the transformer encoder layers. The emb_size determines the dimensionality of these patch embeddings. Larger values of emb_size can capture more fine-grained details, but they also increase the model's computational requirements.

*b) **Testing Analysis**:* As shown in Figure 9, the model starts with relatively low accuracy and high loss values, indicating that it initially struggles to make accurate predictions. However, as training progresses, both accuracy and

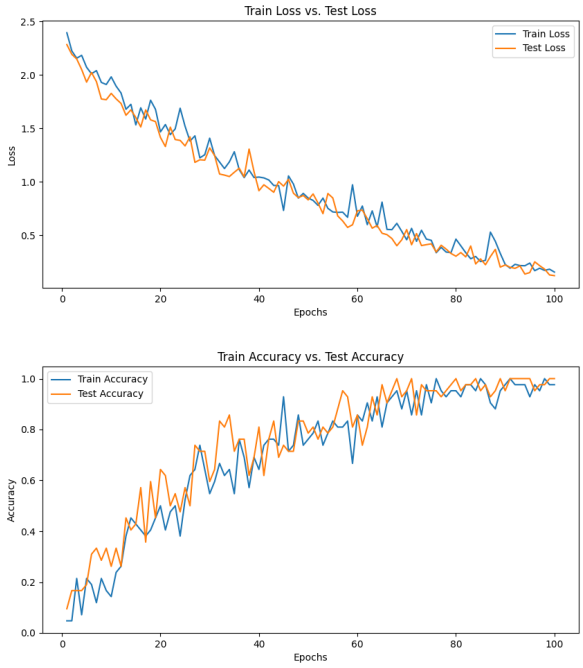


Fig. 9. The result of Modified ViT

loss improve, suggesting that the model is learning to better represent and classify the data.

The model seems to converge as the training progresses, with accuracy reaching high values and loss decreasing. This suggests that the model is learning meaningful representations of the input data and is capable of making accurate predictions.

There are some fluctuations in both accuracy and loss values during training. This could be due to the complexity of the dataset or variations in the training data. Techniques such as regularization or adjusting the learning rate might help in reducing these fluctuations.

C. Experiment III: Fine Tune

In the Fine Tune experiment, a set of different parameters was employed compared to the previous experiments. Preprocessing techniques were applied to the data, as indicated by the "preprocess" parameter set to True. Regarding the architectural parameters, the values of depth and emb_size were adjusted. In this experiment, the depth parameter was set to 3, indicating that the model consisted of three transformer encoder layers. By increasing the depth, the model can become capable of capturing more complex patterns and dependencies within the EEG data. However, it is important to note that increasing the depth also leads to higher computational complexity and memory requirements.

Similarly, the emb_size parameter was set to 10, representing the embedding size or dimensionality of the patch embeddings in the model. By choosing a larger emb_size, the model can capture more fine-grained details and information within the EEG signals. However, this also increases the computational demands of the model. The parameters used for this experiment are shown in Table V.

TABLE V
THE PARAMETERS OF FINE TUNE

PARAMETER	TYPE	VALUE
dataset	null	BCI Competition IV Dataset 2a
channel	int	22
epoch	int	100
learining-rate	float	0.0002
preprocess	bool	True
depth	int	3
emb_size	int	10

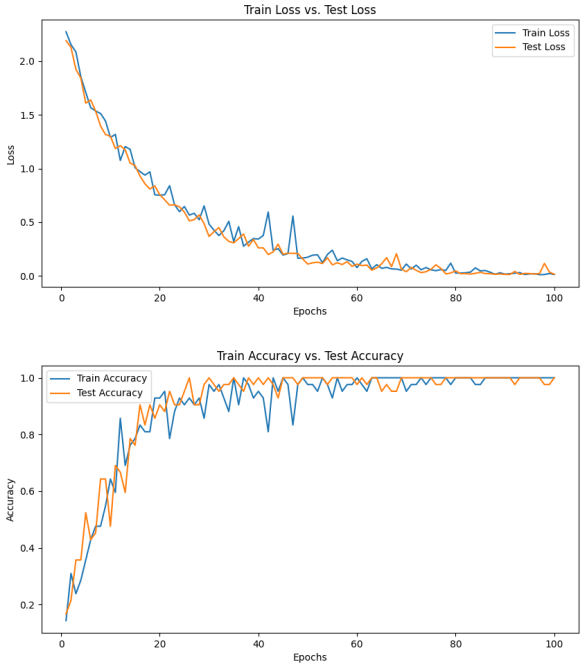


Fig. 10. The result of fine tune

*c) **Test Analysis**:* As shown in Figure 10, the test accuracy is higher than the previous log. In the previous log, the test accuracy reached 0.9762, while in this log, the test accuracy reached a perfect score of 1.0. This indicates that the model is better able to classify EEG data accurately, capturing more intricate patterns in the input.

Comparing the training accuracy with the test accuracy, we can see that the model achieves perfect accuracy on both the training and test sets. This suggests that the model has learned the training data well and is generalizing effectively to unseen data, indicating that there is no significant overfitting.

D. Experiment IV: PhysioNet EEG Motor Movement/Imagery Dataset

In this experiment, the PhysioNet EEG Dataset was used to validate the performance of our approach. The specific number of channels used in this test ranged from 16 to 23, indicating a narrower channel range compared to the previous experiments. The model was trained for 100 epochs with a learning rate of 0.0002.

Regarding the architectural parameters, the values of depth and emb_size were set to 2 and 5, respectively. The depth parameter indicates that the model consisted of two transformer encoder layers, allowing it to capture patterns and

TABLE VI
THE PARAMETERS OF PHYSIONET DATASET

PARAMETER	TYPE	VALUE
dataset	null	PhysioNet Dataset
channel	int	16-23
epoch	int	100
learining-rate	float	0.0002
preprocess	bool	False
depth	int	2
emb_size	int	5

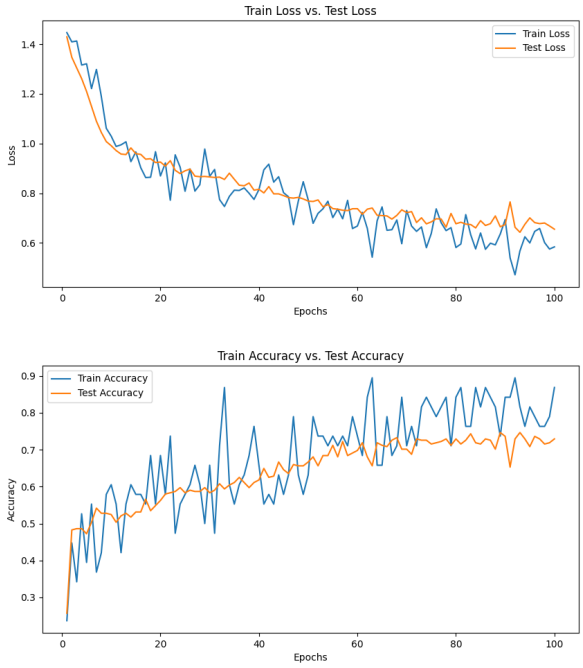


Fig. 11. The result of the PhysioNet Dataset

dependencies within the EEG data. The emb_size parameter represents the dimensionality of the patch embeddings in the model. In this case, the emb_size was set to 5, which determines the size of the lower-dimensional space to which the input patches are projected. A smaller emb_size value reduces the computational requirements of the model but may limit its ability to capture fine-grained details.

*d) **Test Analysis**:* In Figure 11, the training loss values range from 1.4465 to 0.4712. Initially, the loss is relatively high but decreases with each epoch, indicating that the model is learning and adjusting its parameters to minimize the error. The decreasing trend suggests that the model is converging. However, it is important to note that the training loss continues to decrease even when the test accuracy starts to fluctuate, which could be a sign of overfitting.

The test loss values range from 1.4296 to 0.6425. Similar to the training loss, the test loss initially starts high and gradually decreases with each epoch. However, it also exhibits fluctuations, particularly towards the end of the training process. This could be an indication of the model's inability to generalize well to unseen data.

The training accuracy values range from 0.2368 to 0.8947 (with precision ranges from 0.2474 to 0.9023 and recall ranges from 0.2345 to 0.8892). The model shows improvement in accuracy throughout the training process, but the final accuracy is not perfect. This suggests that the model has learned to fit the training data to some extent. However, the model may not have achieved optimal performance and may still have room for improvement. The test accuracy values range from 0.2569 to 0.7465 (with precision ranges from 0.2667 to 0.7723 and recall ranges from 0.2453 to 0.0.7334). Initially, the accuracy is low but gradually improves over time, reaching a peak at 0.7465. However, there are fluctuations during training, suggesting potential instability in the model's performance. It is worth noting that the accuracy does not consistently increase with each epoch, indicating that the model may struggle to generalize to unseen data.

E. Comparison with Relevant Approaches

Due to the popularity of smart home, smart city and the fast-developing IoT scenarios, EEG authentication has received much attention from both academia and industry. Here we provided a comparison with some typical and relevant methods in the literature.

TABLE VII
PERFORMANCE COMPARISON WITH BASELINE AND SIMILAR METHODS WITH BCI IV 2A DATASET AND PHYSIONET DATASET

Method / Mean Accuracy (%)	BCI IV 2a	PhysioNet
Lawhern et al. [13]	94.33	88.83
Olivas-Padilla et al. [16]	92.54	91.53
Xu et al. [23]	96.53	88.64
Du et al. [6]	97.32	90.76
Hu et al. [11]	97.11	91.88
Zeynali et al. [25]	96.83	91.84
Ouyang et al. [17]	97.13	91.23
Our method	97.65	91.81

For instance, Lawhern et al. [13] introduced a lightweight CNN model for EEG classification. Olivas-Padilla et al. [16] also introduced a CNN model for multiple motor imagery classification. Xu et al. [23] used a Wavelet Transform with CNN to classify the EEG signals. Du et al. [6] presented an EEG Temporal-Spatial Transformer (ETST) model for accurate personal identification. Hu et al. [11] compbined AuthConformer and Convolutional transformer model to perform continuous authentication based on users' behavioral actions. Zeynali et al. [25] presented a hybrid scheme with deep learning and ensembles for EEG classification and Ouyang et al. [17] introduced a SiamEEGNet model, combined EEGNet and Siamese networks for EEG authentication.

Table VII presents the comparison results. It is found that for the BCI IV 2a dataset, our method could achieve the best mean accuracy. Some hybrid schemes could achieve a similar performance such as the methods from Du et al. [6], Hu et al. [11] and Ouyang et al. [17]. While for the PhysioNet dataset, the method from Hu et al. [11] could achieve the best accuracy; however, the value was very close among our method, Zeynali et al. [25], Olivas-Padilla et al. [16] and Ouyang et al. [17]. It is worth noting that most relevant studies were using a hybrid scheme, so the performance of our method (only using an improved Transformer) is very encouraging and has a great potential.

F. Limitations and Discussion

Performance and results. We conducted several experiments to evaluate the performance of our model and the original model for EEG-based authentication. We tested the BCI Competition IV Dataset 2a and the PhysioNet Dataset with different parameters.

The first experiment on the original model utilized a transformer architecture with various data augmentation methods, including subsampling, random cropping, and CSP. The model achieved a training accuracy of 0.95 and a test accuracy of 0.6955. Although the model showed stable performance, it reached a plateau, indicating limited improvement beyond a certain point.

We then experimented with our developed and optimized ViT model. The results demonstrated that our model could improve the performance compared to the original model. Our model achieved a training accuracy of 1.0 and a test accuracy of 0.7504. However, fluctuations in the test accuracy suggested the need for further optimization. Next, we fine-tuned the model by applying preprocessing techniques and adjusting the depth and embedding size. The fine-tuned model achieved perfect train accuracy of 1.0 and significantly improved test accuracy of 0.9908. This indicated that our model was able to accurately authenticate users based on EEG data with high generalization capability.

To validate the model's performance, we conducted another experiment with the PhysioNet EEG Motor Movement/Imagery Dataset. The model achieved a training accuracy of 0.8947 and a test accuracy of 0.7465. As it is a large dataset, we can say the results were satisfied. However, fluctuations in accuracy and loss values suggested potential stability and generalization issues in our future work.

Threat model. In this work, we consider a threat model: attackers are assumed to have the same prior knowledge and behavioral abilities as normal users. Similar to prior work [22], we consider two types of attacks: insider attack and outsider attack. For an insider attack, the attacker is one of the system users with template enrolled, while trying to impersonate other users. For an outsider attack, the attackers come from outside the registered users with no template enrolled.

In many existing research studies on EEG authentication, attacks are not often considered and it is an open challenge in this field. Also, some new attacks are developed, for example, Neupane et al. [15] presented an attack called PEEP, which can passively monitor sensitive typed input, specifically numeric PINs and textual passwords, by analyzing the corresponding neural signals. In this work, we obtained an initial equal error rate (EER) under two attacks: around 1% for insider attack and around 4.5% for outsider attack, which are acceptable. We plan to perform a detailed security analysis in our future work by leveraging new collected data and studying particular attacks such as mimic attack [28] and PEEP.

Parameter selection. As stated in Experiment II, we selected a small value of 0.0002. In the experiment, we have tested increasing or decreasing the learning rate, but found that this value of 0.0002 could deliver the best performance in this model. This suggests that, given the specific structure of the model and the characteristics of the data, this learning rate effectively balances training speed and convergence stability. Our future work could explore dynamic learning rate strategies (such as cosine annealing or adaptive learning rate optimizers) to investigate whether further performance improvements can be achieved.

Metric adoption. In many current research on biometric and EEG authentication, accuracy is the most commonly used metric, e.g., [29]. While many other metrics can be considered to provide a better understanding of the scheme performance, such as precision, recall, confusion matrice, etc. In our future work, we plan to consider more datasets and adopt more metrics to verify the obtained results.

Computing complexity. The training method runs 1000 epochs with batch size of 50 for each subject (22 subjects in total), hence the empirical execution times mainly depends on hardware. Our code is publicly available, so the time can be computed based on specific configurations. For the Analytical Complexity, we know that: i) Self-attention mechanism: Quadratic in input sequence length $O(n^2)$ where n is the number of patches or tokens processed, and ii) Feedforward layers: Linear with embedding size $O(d)$ where d is the dimension of embeddings. Overall complexity per transformer block is $O(n^2 d)$, dominated by self-attention.

To further enhance the adoption of EEG authentication, we can consider some practical implementations. For instance, Cabarcos et al. [4] introduced three more effective authentication tasks via cognitive semantic processing based on consumer devices, which can complement our proposed method and most existing EEG authentication methods. Also, we need to consider more usable aspect, e.g., Rose et al. [31] developed NeuroPack–a Python library tailored EEG authentication system and explore various usability questions such as how users perceive the usability of brainwave-based authentication in the real world, and under what conditions users are willing to use brainwave based authentication.

VI. Conclusion

In the era of IoMT and Metaverse, more data would be transferred among IoT devices. To protect the security and privacy of stored sensitive data, EEG-based authentication is a natural and necessary security mechanism. In this work, we investigated the potential of Transformer model in the domain of EEG-based authentication. Our goal was to address the limitations of traditional learning models in the aspect of effectively capturing the complex spatial and the temporal dependencies present in EEG signals.

We delved into the Transformer architecture and its application to EEG-based authentication. To better capture dependencies in the input sequence, we developed and optimized a ViT model tailored for EEG data, incorporating channel attention and convolutional layers to handle spatial and temporal dependencies. In the evaluation, we evaluated the performance of our proposed ViT model compared with the original model and similar methods. Our findings showed that our proposed Transformer model could hold great promise for EEG-based authentication. One of its key advantages lies in its ability to effectively capture long-range dependencies, crucial for

analyzing EEG signals with intricate temporal and spatial relationships. We found that the Transformer's self-attention mechanism allows for a holistic understanding of EEG data, leading to the improved authentication performance.

REFERENCES

[1] BCI Competition 2008 C Graz data set A. (accessed on 1 January 2024) https://lampx.tugraz.at/~bci/database/001-2014/description.pdf

[2] H. Aurlien, I.O Gjerde, J.H Aarseth, G. Eldoen, B. Karlsen, H. Skeidsvoll, and N.E Gilhus, "EEG background activity described by a large computerized database," *Clinical Neurophysiology* 115(3), pp. 665-673, 2004.

[3] A.J. Bidgoly, H.J. Bidgoly, and Z. Arezoumand, "A survey on methods and challenges in EEG based authentication," *Comput. Secur.* vol. 93, 101788, pp. 1-16, 2020.

[4] P.A. Cabarcos, T. Habrich, K. Becker, C. Becker, and T. Strufe: Inexpensive Brainwave Authentication: New Techniques and Insights on User Acceptance. *USENIX Security Symposium* 2021: 55-72.

[5] S. Cheng, J. Wang, D. Sheng, and Y. Chen, "Identification With Your Mind: A Hybrid BCI-Based Authentication Approach for Anti-Shoulder-Surfing Attacks Using EEG and Eye Movement Data," *IEEE Trans. Instrum. Meas.* 72, pp. 1-14, 2023.

[6] Y. Du, Y. Xu, X. Wang, L. Liu, and P. Ma, "EEG temporal-spatial transformer for person identification," *Sci Rep.* 12(1), 14378, 2022.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *ICLR*, pp. 1-21, 2021.

[8] F. Fahimi, S. Dosen, K.K. Ang, N. Mrachacz-Kersting, and C. Guan, "Generative Adversarial Networks-Based Data Augmentation for Brain-Computer Interface," *IEEE Trans. Neural Networks Learn. Syst.* 32(9), pp. 4039-4051, 2021.

[9] C.A. Fidas and D.P. Lyras, "A Review of EEG-Based User Authentication: Trends and Future Research Directions," *IEEE Access* 11, pp. 22917-22934, 2023.

[10] A. Hassanpour, M. Moradikia, H. Adeli, S.R. Khayami, and P. Shamsinejadbabaki, "A novel end-to-end deep learning scheme for classifying multi-class motor imagery electroencephalography signals," *Expert Syst. J. Knowl. Eng.* 36(6), 2019.

[11] M. Hu, K. Zhang, R. You, and B. Tu, "AuthConFormer: Sensor-based Continuous Authentication of Smartphone Users Using A Convolutional Transformer," *Comput. Secur.* 127: 103122, 2023.

[12] Z. Liao, X. Pang, J. Zhang, B. Xiong, and J. Wang, "Blockchain on Security and Forensics Management in Edge Computing for IoT: A Comprehensive Survey," *IEEE Trans. Netw. Serv. Manag.* 19(2), pp. 1159-1175, 2022.

[13] V.J. Lawhern, A.J. Solon, N.R. Waytowich, S.M. Gordon, C.P. Hung, and B.J. Lance, "EEGNet: a compact convolutional neural network for EEG-based brainCcomputer interfaces," *Journal of Neural Engineering* 15(5), 056013, 2018.

[14] W. Lu, T.P. Tan, and H. Ma, "Bi-Branch Vision Transformer Network for EEG Emotion Recognition," *IEEE Access* 11, pp. 36233-36243, 2023.

[15] A. Neupane, M.L. Rahman, and N. Saxena, "PEEP: Passively Eavesdropping Private Input via Brainwave Signals," *Financial Cryptography* 2017: 227-246.

[16] B.E. Olivas-Padilla and M.I.C. Murguia, "Classification of multiple motor imagery using deep convolutional neural networks and spatial filters," *Appl. Soft Comput.* 75, pp. 461-472, 2019.

[17] R. Ouyang, X. Wu, and Z. Lv, "Personal Identification and Authentication in Multi-Task EEG Database Using EEGNet and Siamese Network," in *IJCNN*, pp. 1-8, 2024.

[18] Y.A. Qadri, A. Nauman, Y.B. Zikria, A.V. Vasilakos, S.W. Kim, "The Future of Healthcare Internet of Things: A Survey of Emerging Technologies," *IEEE Commun. Surv. Tutorials* 22(2), pp. 1121-1167, 2020.

[19] Z. Tayeb, J. Fedjaev, N. Ghaboosi, C. Richter, L. Everding, X. Qu, Y. Wu, G. Cheng, and J. Conradt, "Validating Deep Neural Networks for Online Decoding of Motor Imagery Movements from EEG Signals," *Sensors* 19(1), 210, 2019.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All you Need," in *NIPS*, pp. 5998-6008, 2017.

[21] Y. Wang, Z. Su, N. Zhang, R. Xing, D. Liu, T.H. Luan, and X. Shen, "A Survey on Metaverse: Fundamentals, Security, and Privacy," *IEEE Commun. Surv. Tutorials* 25(1), pp. 319-352, 2023.

[22] B. Wu, W. Meng, and W.Y. Chiu, "Towards Enhanced EEG-based Authentication with Motor Imagery Brain-Computer Interface," in *ACSAC*, pp. 799-812, 2022.

[23] B. Xu, L. Zhang, A. Song, C. Wu, W. Li, D. Zhang, G. Xu, H. Li, and H. Zeng, "Wavelet Transform Time-Frequency Image and Convolutional Network-Based Motor Imagery EEG Classification," *IEEE Access* 7, pp. 6084-6093, 2019.

[24] T. Xu, H. Wang, G. Lu, F. Wan, M. Deng, P. Qi, A. Bezerianos, C. Guan, and Y. Sun, "E-Key: An EEG-Based Biometric Authentication and Driving Fatigue Detection System," *IEEE Trans. Affect. Comput.* 14(2), pp. 864-877, 2023.

[25] M. Zeynali, H. Seyedarabi, and R. Afrouzian, "Classification of EEG signals using Transformer based deep learning and ensemble models," *Biomed. Signal Process. Control.* 86 (Part A): 105130, 2023.

[26] P. Guo, W. Liang, and S. Xu, "A privacy preserving four-factor authentication protocol for internet of medical things," *Comput. Secur.* 137: 103632, 2024.

[27] J. Wu, W.Y. Chiu, and W. Meng, "KEP: Keystroke Evoked Potential for EEG-Based User Authentication," in *AIS&P*, pp. 513-530, 2023.

[28] W.Y. Chiu, W. Meng and W. Li, "I Can Think Like You! Towards Reaction Spoofing Attack on Brainwave-based Authentication," in *SpaCCS*, pp. 251-265, Springer, 2020.

[29] A. Casanova, L. Cascone, A. Castiglione, W. Meng, and C. Pero, "User Recognition based on Periocular Biometrics and Touch Dynamics," *Pattern Recognition Letters*, vol. 148, pp. 114-120, 2021.

[30] J. Ju and H. Li, "A Survey of EEG-Based Driver State and Behavior Detection for Intelligent Vehicles," *IEEE Trans. Biom. Behav. Identity Sci.* 6(3), pp. 420-434, 2024.

[31] M. Rose, E. Kablo, and P.A. Cabarcos, "Overcoming Theory: Designing Brainwave Authentication for the Real World," in *EuroUSEC* 2023: 175-191.

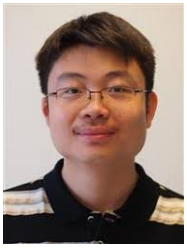
[32] Y. Wang, B. Zhang, and L. Di, "Research Progress of EEG-Based Emotion Recognition: A Survey," *ACM Comput. Surv.* 56(11), pp. 288:1-288:49, 2024.

[33] S. Cheng, J. Wang, D. Sheng, and Y. Chen, "Identification With Your Mind: A Hybrid BCI-Based Authentication Approach for Anti-Shoulder-Surfing Attacks Using EEG and Eye Movement Data," *IEEE Trans. Instrum. Meas.* 72, pp. 1-14, 2023.

[34] H. Zhu, M. Xiao, D. Sherman, and M. Li, "SoundLock: A Novel User Authentication Scheme for VR Devices Using Auditory-Pupillary Response," in *NDSS* 2023.

[35] W. Meng, Y. Cai, L.T. Yang, and W.Y. Chiu, "Hybrid Emotion-aware Monitoring System based on Brainwaves for Internet of Medical Things," *IEEE Internet of Things Journal* 8(21), pp. 16014-16022, IEEE, 2021.

Chunxue Li obtained her Master degree in 2023 from the Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), Denmark. Her primary research interests are Artificial Intelligence and Security, especially brainwave-based authentication.



Weizhi Meng is currently a Full Professor in the School of Computing and Communications, Lancaster University, United Kingdom. Prior to that, he was a tenured faculty in DTU Compute, Technical University of Denmark. He obtained his Ph.D. degree in Computer Science from the City University of Hong Kong, Hong Kong SAR. He was a recipient of the Hong Kong Institution of Engineers (HKIE) Outstanding Paper Award for Young Engineers/Researchers in both 2014 and 2017. He also received the IEEE ComSoc Best Young Researcher Award for Europe, Middle East, & Africa Region (EMEA) in 2020. His primary research interests are cyber security and intelligent technology in security, including blockchain, intrusion detection, AI security, IoT security, biometric authentication, and trust management. He serves as associate editors / editorial board members for many reputed journals such as IEEE TDSC, as well as chair for various international conferences such as ACM CCS'23 and ESORICS'22. He is a senior member of IEEE.

Wenjuan Li obtained the Ph.D degree from the Department of Computer Science, City University of Hong Kong (CityU). She received both Research Tuition Scholarships and Outstanding Academic Performance Award during her doctorate studies. She is currently an Assistant Professor in the Department of Mathematics and Information Technology, The Education University of Hong Kong. Prior to that, she was a Research Assistant Professor in the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, China. Her research interests include network management and security, intrusion detection, spam detection, trust management, blockchain security, and E-commerce security. She is a senior member of IEEE.