M-MNet for Cancer Classification by Constructing Somatic Mutation Map

Chenxu Quan[1,†], Xin Chen[1,†], Fenghui Liu[2], Lin Qi[1] and Yun Tie[1,*]

[1]*Zhengzhou University, Zhengzhou, Henan, 450000, China*

[2]*The First Affiliated Hospital of Zhengzhou University, Zhengzhou, Henan, 450000, China*

Abstract

Due to the high fatality rate of cancer, timely detection and treatment in the early stage of cancer is very important. Cancer classification studies based on somatic mutation data are helpful for physicians to identify cancer types at the genetic level and reduce the possibility of misdiagnosis and missed diagnosis. However, the one-dimensional (1-D) high redundancy of somatic mutation data resulted in the low robustness and overfitting of the model. In addition, current models based on convolution neural networks (CNNs) fail to take global features of input data into account and are inferior in classification performance. In this paper, we proposed a gene mutation map construction method to realize the dimension transformation of somatic mutation data and make it suitable for existing image classification models, which are based on the RGB three-channel principle of the image. Then, based on the prediction results of driver genes, the feature selection optimization (FSO) algorithm is performed on the original mutation map to solve the problems of high noise and sparsity of the original mutation map. Furthermore, a classification network named M-MNet is introduced based on inverted residual module and multi-head self-attention module. The experimental results show that the proposed method have improved the overall classification performance, and the overall method achieves 94.62% accuracy and 94.34% f1 score in cancer classification tasks of 19 tumor cohorts, which has good cancer classification ability.

Keywords

Somatic mutation, Driver genes, Cancer classification, Feature selection optimization, M-MNet

1. Introduction

There are many types of mutations in the somatic cell genome, such as Single Nucleotide Variants (SNVs), Insertions and Deletions (InDels), and chromosome Structural Variations (SVs) [1]. When specific genomic mutations occur in somatic cells, they can promote the development and malignant proliferation of cancer cells. These genes are known as driver genes. For example, it has been confirmed that mutations in the VHL and MET genes lead to kidney cancer [2]. Depending on the mechanisms by which they induce cancer, driver genes can be classified as oncogenes or tumor suppressor genes, both of which interact to maintain stable positive and negative regulatory signals. Oncogenes are often expressed at low levels or not expressed in the genome. When they undergo mutation and abnormal activation, they become carcinogenic factors that induce cancer. Tumor suppressor genes, on the other hand, are genes in the genome that have inhibitory effects on cell growth and potential anti-cancer effects. Mutations or inactivation of tumor suppressor genes can lead to cell carcinogenesis. However, not all mutated genes are driver genes. There are neutral mutations in the human body that do not promote cell carcinogenesis, and genes that undergo such mutations are called passenger genes.

Cancer driver genes play a significant role in various clinical aspects of cancer prevention, early detection and diagnosis, staging and classification, as well as rehabilitation treatment. Both driver genes and passenger genes are mutated genes, but they have distinct roles in regulating cellular physiological mechanisms and their pathological analysis and treatment. Therefore, it is of great significance to identify driver genes among all mutated genes in tumor cells. Due to the high complexity

of cancer gene mutation mechanisms [3], it is challenging to identify the distinguishing features that can effectively differentiate driver genes from passenger genes. Accurately identifying driver genes remains a huge challenge. In recent years, with the rapid development of computer science and the emergence of second-generation sequencing technologies, such as high-throughput sequencing, it has become possible to analyze cancer driver genes with the support of big data. This data-driven research approach significantly improves the efficiency of cancer research. Taking advantage of these advancements, many complex computational methods have been proposed for detecting cancer driver gene mutations and conducting an in-depth analysis of the regulatory mechanisms behind driver genes in cancer.

This article focuses on the prediction of driver genes and cancer classification tasks based on somatic mutation data, aiming to address the challenges encountered in driver gene prediction and cancer classification. Among them, the driver gene prediction methods based on mutation position face the following problems: (1) The statistical determination of mutation probabilities for nucleotide contexts is independent, which can lead to overfitting of background mutation signals and overlook important signals conveyed by low-frequency mutation sites. (2) The computational complexity increases exponentially when expanding nucleotide contexts. (3) Traditional clustering algorithms such as K-means and DBSCAN are no longer suitable for handling complex mutation data, resulting in poor clustering performance.

The main contributions of this study to address the aforementioned problems are as follows:

1) Firstly, this study applies the RGB three-channel principle to perform dimensionality transformation on somatic mutation data, resulting in a two-dimensional gene mutation image data. This enables the use of existing image classification models for somatic mutation data.

2) Secondly, this study employs a feature selection optimization algorithm to address the issue of high noise and sparsity in the two-dimensional mutation image. This optimization work in feature gene selection effectively enhances the model's generalization ability.

3) Lastly, to better capture both local and global features of the images, this study proposes the M-MNet network model by combining the inverted residual module with the multi-head self-attention module. Specifically, the inverted residual module is utilized to extract local features, followed by the multi-head self-attention module to capture global features. This enables accurate feature extraction and classification of mutation images.

2. Related Work

The computational methods used for cancer driver gene discovery mainly include three main classes: methods for identifying individual cancer driver genes, methods for identifying cancer driver gene modules, and methods for discovering personalized cancer driver genes (i.e., driver genes specific to individual patients).

Single cancer driver gene identification can be divided into two subcategories based on the key techniques used in the methods: mutation-based methods and subnetwork-based methods. Mutation-based methods utilize various features of mutations, such as the significance, functional impact, location, and other information, to discover cancer driver genes. Methods based on the significance of gene mutations include MuSiC [4] and MutSigCV [5]. Methods based on the functional impact of gene mutations include OncodriveFM [6], OncodriveFML [7], DriverML [8], and others. These methods often select driver gene candidates based on mutations in genes that have a significant functional impact rather than evaluating the number of mutations. Therefore, these methods can often detect low-frequency mutations that play important roles in cancer development. Methods based on the location information of gene mutations usually rely on clustering methods and are often referred to as hotspot-based methods [9]. Hotspots typically refer to high-density mutation regions, often driven by positive selection, and are commonly found in functionally important domains or residues in the

three-dimensional protein structure [10]. The OncodriveCLUST method [11] is a typical representative of hotspot methods. The MMC algorithm [12] adjusts the influence weights of mutation sites on surrounding sites through kernel density estimation for clustered identification of mutation genes. The clustering algorithm HotMAPS [13] considers more hotspot information in the 3D protein space by combining tumor mutation data with PDB data. Additionally, MutPanning [14] identifies driver factors based on the ratio of non-synonymous substitution rate to synonymous substitution rate (dn/ds). Network-based methods often predict cancer driver genes by evaluating the role of genes in biological networks and combining them with gene mutation information. DriverNet [15] reveals cancer driver genes by assessing the impact of mutations on cancer transcriptional networks.

Identification of cancer driver modules. The majority of methods for identifying cancer driver modules are based on the mutual exclusivity of mutations. CoMEt [16] uses mutual exclusivity techniques to detect cancer driver modules. Similarly, WeSME [17] evaluates the mutual exclusivity of gene mutations to detect cancer driver genes. However, WeSME does not evaluate genes within the same pathway but only considers mutually exclusive pairs of mutated genes as candidate cancer driver genes for modularization.

Personalized cancer driver gene identification. Personalized cancer driver gene identification methods are based on gene regulatory networks to recognize driver genes. For example, PNC [18] identifies the minimum gene set that covers all edges in a bipartite graph as cancer driver genes.

3. Methods

3.1. Data Dimension Transformation

Different from traditional 1-D data, gene mutation data has randomness and low frequency. The mutation information randomly distributed in the long redundant gene text sequence is difficult to be effectively captured by the existing deep learning model, and the long data length requires high complexity and performance of the model, and it is difficult to effectively extract features to realize classification tasks by conventional means. Thus, we transform the dimension of somatic mutation data based on the RGB three-channel image principle to be suitable for training of deep learning model.

Figure 1 illustrates the construction process of the gene mutation map. First, count the mutated genes in each tumor cohort of the dataset and sort them based on their chromosomal positions (chromosome $1, 2, \cdots, 22$, X and Y). The mutated genes in the i-th chromosome of j-th tumor cohort is arranged to a matrix S_{ij}, where $0 \leq i \leq 30$ and $0 \leq j \leq 18$.

Next, the mutated genes from different tumor cohorts but the same chromosome are resorted. At this stage, the length of the set $S_{i\cdot} = S_{i0} \cap S_{i1} \cap \cdots \cap S_{i18}$, which consists of mutated genes from different tumor cohorts in i-th chromosome, is L_i. Assuming the constructed gene mutation graph has a shape of $N \times N$, at this point, the i-th chromosome occupies m_i rows in the mutation graph, where:

$$m_i = \begin{cases} L_i/N + 1, & L_i \bmod N \neq 0 \\ L_i/N, & L_i \bmod N = 0 \end{cases} \tag{1}$$

Then all the genes on the chromosomes occupy M rows, where M is defined as:

$$M = \sum_{i=0}^{23} m_i = \sum_{0}^{23} \begin{cases} L_i/N + 1, & L_i mod N \neq 0 \\ L_i/N, & L_i mod N = 0 \end{cases}, M \leq N \tag{2}$$

According to the above definition, we can build a gene mutation map of size $N \times N$, which includes all mutated genes from the 19 tumor cohorts. In this article, the replacement, insertion, and deletion genes from single nucleotide variants are statistical and correspond to the RGB channels of the image, respectively. To map mutation information onto an image, this method first selects the maximum mutation value, $Max_{mutation}$, from all samples in the corresponding mutation type. $Max_{mutation}$ is then used as one of the indicators to map the mutation quantity to grayscale values. For convenience,
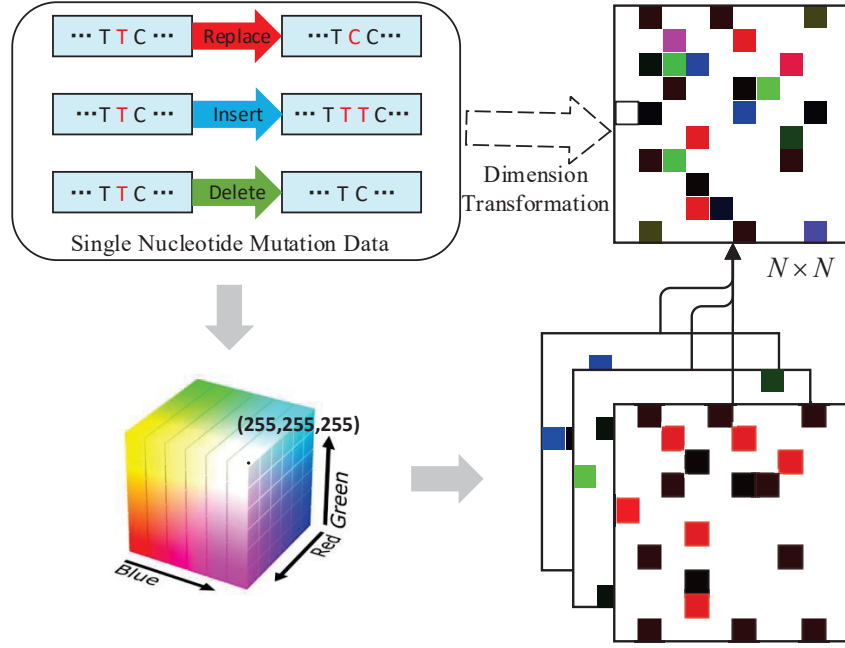
Figure 1: The construction process of gene mutation map.

assume that a single nucleotide of mutation gene G makes a replacements in the sample gene fragment. And its value in the red channel is b. The relationship between a and b is:

$$\frac{b}{255} = \frac{a}{Max_{mutation}}, b \leq 255, a \leq Max_{mutation} \tag{3}$$

According to the above definition, the conversion from mutation quantity to grayscale values in a single-channel image can be accomplished. Finally, the single-channel images corresponding to each mutation type in the samples are merged to obtain a two-dimensional mutation image.

3.2. Feature Selection Optimization

The high noise and sparsity issues present in mutation profile data hinder the effective extraction of feature information. Overlearning the noise in the data can increase the complexity of the model and lead to overfitting. To avoid interference caused by high noise and sparsity in the data, this method narrows down the scope of feature selection based on previous work [19]. It employs clustering algorithms to filter candidate driver genes and potential driver genes for feature selection optimization (FSO). Passenger genes that have no significant relevance to tumor formation or development are eliminated. This approach helps improve the overall training efficiency and classification performance of the model.

Assuming that the dimension of the gene mutation map after FSO is $N' \times N'$, a' represents the number of selected feature genes for each tumor cohort. According to the principles of constructing the mutation map, there should be a certain relationship between a' and N' denoted by $24N' + 19a' - 24 \leq N'^2$.

This is because, in the dimension transformation process proposed in this paper, each row of the mutation map corresponds to the mutation information on one chromosome. Since the distribution of mutated genes across the 24 chromosomes in the human body is not uniform, if there $N' + 1$ mutated genes in one chromosome, that chromosome will occupy two rows in the mutation map. To better adapt to the training process of deep learning models, this paper sets N' to be 56. The maximum integer value that a' can take is theoretically 95. However, in practice, the probability of the occurrence of this situation (where $N' + 1$ mutated genes occupy two rows) is almost zero. Therefore, in this paper, a' is set to 100. This means that for each tumor cohort, the selected mutated genes are the top 100 ranked genes according to the clustering prediction results. If the number of candidate driver genes is less than 100, potential driver genes will be further selected. After FSO, the dimension of the gene

Figure 2: The network structure of M-MNet.

mutation matrix is reduced to 56×56, effectively addressing the high noise and sparsity issues in the original dimension transformation method. This is because, in the dimension transformation process proposed in this paper, each row of the mutation map corresponds to the mutation information on one chromosome. Since the distribution of mutated genes across the 24 chromosomes in the human body is not uniform, if there $N' + 1$ mutated genes in one chromosome, that chromosome will occupy two rows in the mutation map. To better adapt to the training process of deep learning models, this paper sets N' to be 56. The maximum integer value that a' can take is theoretically 95. However, in practice, the probability of the occurrence of this situation (where $N' + 1$ mutated genes occupy two rows) is almost zero. Therefore, in this paper, a' is set to 100. This means that for each tumor cohort, the selected mutated genes are the top 100 ranked genes according to the clustering prediction results. If the number of candidate driver genes is less than 100, potential driver genes will be furt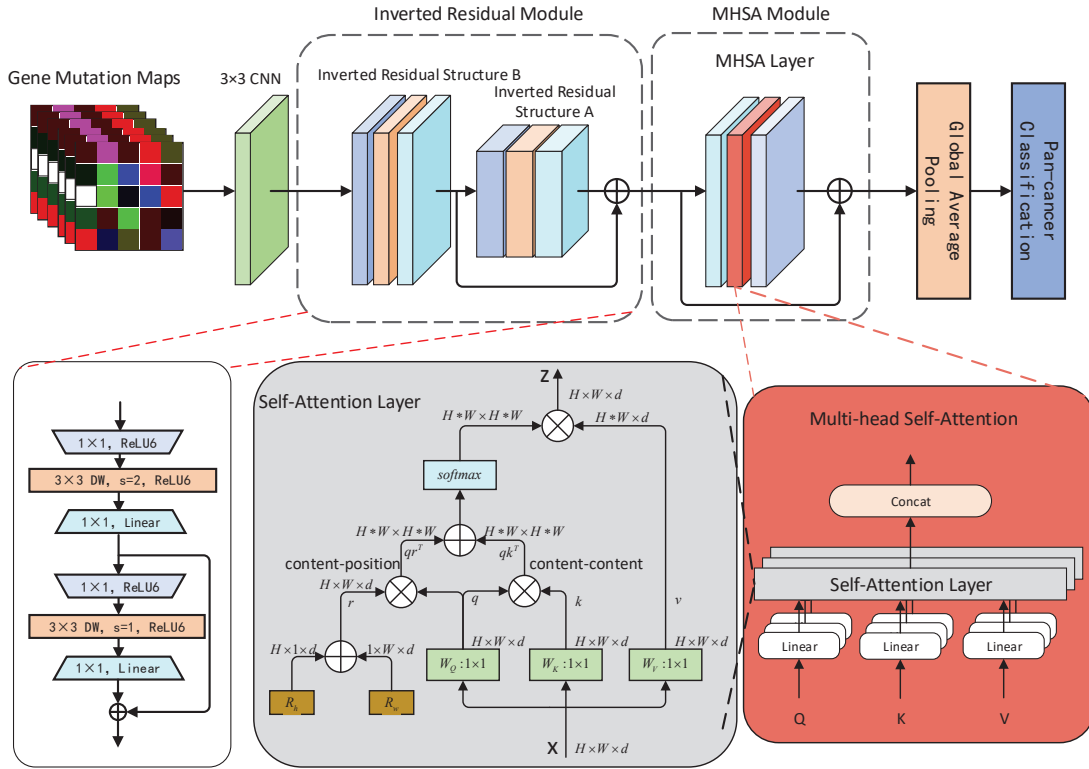her selected. After FSO, the dimension of the gene mutation matrix is reduced to 56×56, effectively addressing the high noise and sparsity issues in the original dimension transformation method.

3.3. M-MNet Classification Network

The proposed network model, M-MNet, is based on the inverted residual module and the multi-head self-attention module. The overall architecture of M-MNet is illustrated in Figure 2.

The network takes the gene mutation map as input. It starts with a 3×3 convolutional layer with a stride of 2 to extract shallow features. Then, multiple inverted residual modules are employed to extract high-dimensional features. The inverted residual module consists of two types of inverted residual structures, A and B. In the final stage of the model, several multi-head self-attention modules are introduced to capture global features. Each multi-head self-attention module consists of two 1×1 convolutional layers and a multi-head self-attention layer. Finally, the output is obtained through a 7×7 global average pooling layer.

1) **Inverted Residual Module**: In the conventional residual structure, dimension reduction is performed through a 1×1 convolutional layer before being increased again through another 1×1 convolutional layer. However, in our proposed module, dimension augmentation is performed through a 1×1 convolutional layer, followed by feature extraction through a 3×3 convolutional layer, and finally dimension reduction through another 1×1 convolutional layer. We named this structure the "inverted residual block" because of its reversed order of dimension change. This approach effectively utilizes the feature information from different channels at the same spatial location. It is important to note that while the conventional residual structure uses the $ReLU$ activation function, the inverted residual structure uses the $ReLU6$ activation function as follows:

$$y = ReLU6(x) = \min(ReLU(x), 6) = \min(\max(0, x), 6) \tag{4}$$

Due to the potential loss of information caused by the ReLU function when performing non-linear transformations on low-dimensional features, the inverted residual structure has a lower-dimensional output. Therefore, in the dimension reduction process, the linear activation function is used instead of the ReLU activation function. This choice helps preserve the information and prevent unnecessary loss.

2) **Multi-head Self-attention Module**: The core module of the Transformer network is the Multi-Head Self-Attention (MHSA), which is a specialized form of the multi-head attention mechanism. In MHSA, the inputs $\mathbf{K}$, $\mathbf{V}$, and $\mathbf{Q}$ of the multi-head attention are all hidden state matrices $\mathbf{H} = \mathbb{R}^{d \times L_s}$ of the same input sequence, where d represents the dimension of the hidden state and L_s is the length of the sequence. The Transformer network uses absolute positional encoding to enable the attention mechanism to perceive positional information. However, recent studies have found that relative distance-aware positional encoding is more suitable for visual tasks. This is because attention not only considers the content of information but also takes into account the relative distances between features at different positions, effectively linking information across objects with positional awareness. In this paper, the two-dimensional relative positional encoding self-attention from reference [20] is used to implement the multi-head self-attention mechanism.

In the two-dimensional relative positional encoding attention module, all attention mechanisms are performed on a two-dimensional feature map. The relative distance positional encodings $\mathbf{R}_h$ and $\mathbf{R}_w$ are used to represent the height and width of the feature map, respectively. The attention logarithm is denoted as $\mathbf{q}^{\mathbf{k}^r} + \mathbf{q}^{\mathbf{r}^r}$, where $\mathbf{q}$, $\mathbf{k}$, and $\mathbf{r}$ represent the query, key, and positional encoding (relative distance encoding), respectively. The $\oplus$ and $\otimes$ symbols represent element-wise addition and matrix multiplication, respectively, and 1×1 denotes point-wise convolution. It should be noted that in the Transformer network, the normalization layer used is Layer Normalization (LN), while in our model, the normalization layer used for multi-head self-attention is Batch Normalization (BN). The MHSA block in Transformer includes an output projection, but the MHSA used in this paper does not. Additionally, while Transformer uses a single non-linear activation function in the Feed-forward Network (FFN), this paper uses three non-linear activation functions.

3.4. Label Smoothing Regularization

In classification algorithms, data is typically labeled using hard labels, which are represented in the form of one-hot encoding.

However, when the training data is insufficient to reflect the true distribution of the data, the network model may suffer from overfitting, resulting in decreased generalization ability. To address this issue, this article employs Label Smoothing Regularization (LSR) technique to enhance the robustness of the model.

In label smoothing regularization, a uniform distribution is combined with the technique to constrain the model's predicted results by adding noise to the output. In the label smoothing regularization strategy for multi-class tasks, the one-hot encoded label vector y_i is replaced with a label vector $\hat{y}_i$ as follows:

$$\hat{y}_i = \begin{cases} 1 - \alpha, & i = target \\ \frac{\alpha}{K-1}, & i \neq target \end{cases} \tag{5}$$

Where α s a small hyperparameter typically set to 0.1, and K represents the number of classes. Label smoothing regularization is employed to prevent the model from relying solely on the training data distribution during the training process. By adding noise to the output, the label smoothing technique constrains the model, to some extent, and mitigates overfitting. Additionally, label smoothing can make the clusters of different classes more compact, increasing the inter-class distance and reducing the intra-class distance. This, in turn, enhances the model's generalization ability.

4. Experiments and Discussion

All experiments in this paper were conducted on the Ubuntu 16.04 system with Python 3.9 version under the Anaconda environment. The deep learning library used was TensorFlow. The GPU model used was GeForce RTX 2080 Ti with 12GB memory. During the training process, the relevant parameters remained consistent. The optimizer employed was Momentum-SGD (Stochastic Gradient Descent with Momentum). The number of epochs was set to 150, the initial learning rate was 0.01, the decay factor was 0.5, and the momentum parameter was set to 0.9. Early stopping was implemented to prevent a decrease in validation accuracy as the number of iterations increased. The evaluation metrics for model performance in this study were accuracy, precision, recall, and F1 score.

4.1. Dataset

The experimental dataset was downloaded from the TCGA (The Cancer Genome Atlas), which consists of somatic mutation data from 19 tumor cohorts. The data format is in MAF (Mutation Annotation Format) files, which include various mutation types such as silent, missense, nonsense, splice site, and frameshift insertions/deletions. After obtaining the somatic mutation dataset, this study initially converted the MAF files into TSV (Tab-Separated Values) format. Subsequently, filtering was applied to each cohort dataset to exclude synonymous mutations and minimize false positives. This study focused on single nucleotide variations and excluded structural variations and insertions/deletions from the somatic mutation data.

The final dataset contains a total of 6,906 samples, 236,245 gene elements, and 1,678,190 single nucleotide mutation positions. The gene list from the Cancer Gene Census (CGC) was retrieved from the COSMIC (Catalogue of Somatic Mutations in Cancer) website in April 2022. The genomic coordinates for coding genes were obtained from the ENCODE website, using Gencode version 19.

4.2. Ablation Study

To validate and interpret the effectiveness and necessity of the proposed feature selection optimization algorithm, we conducted classification tasks using three different network models: VGG-16, Inception ResNet-V2, and MobileNet-V2. The input data for these models were gene mutation maps obtained by dimensionality transformation of somatic mutation data. The FSO algorithm was employed to perform feature selection on the two-dimensional gene mutation maps.

Table 1 presents the comparison of classification performance using different network models with gene mutation map construction methods alone and in combination with gene mutation map construction and feature selection optimization algorithm. In Table 1, it can be observed that compared to VGG-16, FSO+VGG-16 achieved an improvement of 6.11 percentage points in accuracy, 5.27 percentage points in precision, and 5.69 percentage points in F1 score. FSO+Inception ResNet-V2 showed a general improvement of 1 percentage point in all metrics compared to the original model. FSO+MobileNet-V2 exhibited a 2 percentage point improvement in all metrics compared to MobileNet-V2 as a whole.
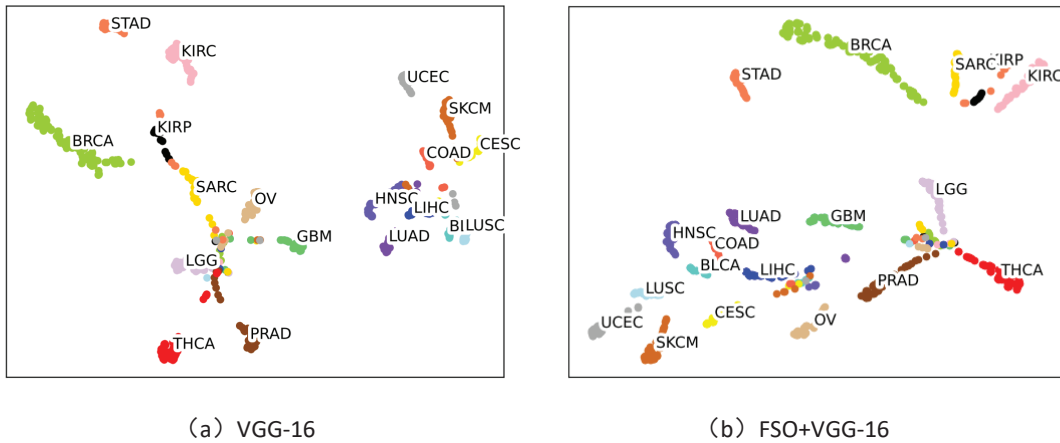
(a) VGG-16   (b) FSO+VGG-16

Figure 3: he visualization of the classification effect of VGG-16 and FSO+VGG-16.

4.3. Feature Visualization

To provide a clearer explanation of the effectiveness of the feature selection optimization module in improving the overall model performance, this subsection utilizes t-SNE (t- distributed Stochastic Neighbor Embedding) to visualize the features extracted by the model. t-SNE compresses the input features before the final classification feature layer into a two-dimensional plane for output, representing a total of 690 gene mutation map samples, to some extent reflecting the model's classification performance.

From Figure 3, it can be observed that after undergoing feature selection optimization, several tumor cohorts such as SARC, OV, and GBM are no longer clustered together with cohorts like LGG, PRAD, and THCA. The classification performance has improved, but there are still some individual samples from cohorts like OV, LIHC, and BRCA that cannot be correctly classified.

In addition, this study also compares the proposed M- MNet model with the previously mentioned best-performing MobileNet-V2 network through visualization. Figure 4 illustrates the classification results of the MobileNet-V2 network and the M-MNet network proposed in this paper for the 690 test samples from the 19 tumor cohorts.

From the figure, it can be observed that the MobileNet- V2 network misclassifies certain samples from tumor cohorts such as PRAD, LIHC, SARC, and COAD into the OV cohort, and it also misclassifies some samples from the CESC and UCEC cohorts into the SKCM cohort. However, in the M-MNet network, this situation is improved, indicating the effectiveness and robustness of the proposed network model.

Table 1

Classification performance comparison of different models on the FSO algorithm.

| Models	Key Performance Indicator (%)			
	Accuracy	Precision	Recall	F1 score
VGG-16	84.82 ± 0.31	85.99 ± 0.42	84.82 ± 0.31	85.40 ± 0.32
FSO+VGG-16	90.93 ± 0.23	91.26 ± 0.34	90.93 ± 0.23	91.09 ± 0.29
Inception ResNet-V2	92.20 ± 0.99	92.37 ± 0.46	92.20 ± 0.99	92.28 ± 0.76
FSO+Inception ResNet-V2	93.35 ± 0.82	93.41 ± 0.31	93.35 ± 0.82	93.38 ± 0.58
MobileNet-V2	91.18 ± 0.53	91.42 ± 0.12	91.18 ± 0.53	91.30 ± 0.33
FSO+MobileNet-V2	93.34 ± 0.56	93.41 ± 0.11	93.34 ± 0.56	93.37 ± 0.23
M-MNet	93.19 ± 0.36	93.11 ± 0.16	93.19 ± 0.36	93.15 ± 0.16
FSO+M-MNet	$\mathbf{94.62 \pm 0.15}$	$\mathbf{94.07 \pm 0.13}$	$\mathbf{94.62 \pm 0.15}$	$\mathbf{94.34 \pm 0.12}$

(a) MobileNet-V2   (b) M-MNet
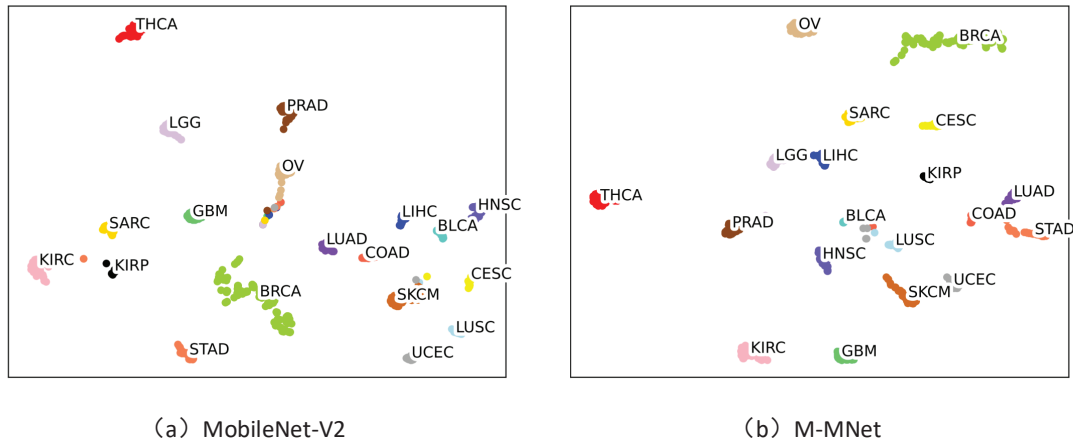
Figure 4: The visualization of the classification effect of MobileNet-V2 and M-MNet.

4.4. Classification performance

We trained the M-MNet network model using gene mutation maps before and after feature selection optimization. We also compared the results with the previous models, and the experimental results are shown in Table 2.

Comparing the experimental results in the fifth and seventh rows of Table 5.3, we can see that the MobileNet-V2 network achieved an accuracy of 91.18%, precision of 91.42%, recall of 91.18%, and F1 score of 91.30%. Compared to the MobileNet- V2 network model with only inverted residual blocks, the M-MNet network showed improvements in all metrics, with an average increase of 2 percentage points. Furthermore, comparing the results in the sixth and eighth rows, it can be observed that FSO+M-MNet outperformed FSO+MobileNet- V2 with an average increase of 1 percentage point, including a 1.28% improvement in accuracy. This demonstrates the effectiveness and necessity of combining the multi-head self- attention module with the inverted residual module. FSO+M- MNet achieved improvements of 1.43, 0.96, 1.43, and 1.19 percentage points in accuracy, precision, recall, and F1 score, respectively, compared to M-MNet. In comparison to the VGG- 16 model, the FSO module had a smaller impact on the overall performance of the M-MNet model, indicating that the feature selection optimization algorithm remains effective for the proposed M-MNet network in this chapter. However, the improvement from the feature selection optimization algorithm becomes smaller when applied to models with better original performance. The gene mutation map construction, feature selection optimization algorithm, and M-MNet network proposed in this paper all contributed to the improvement in model performance, demonstrating the effectiveness and robustness of this approach in cancer classification tasks.

Table 2
Classification performance comparison of different models.

Models	Key Performance Indicator (%)			
	Accuracy	Precision	Recall	F1 score
Huang et al.[21]	90.20 ± 0.31	91.30 ± 0.30	86.20 ± 0.67	93.60 ± 0.19
Szegedy et al.[22]	86.00 ± 0.60	91.30 ± 1.16	67.10 ± 4.23	91.10 ± 0.40
Wang et al.[23]	88.86 ± 0.49	77.85 ± 0.46	91.93 ± 0.99	92.80 ± 0.76
Schlichtkrull et al.[24]	84.02 ± 0.62	87.61 ± 0.39	71.14 ± 0.72	91.57 ± 0.68
Dosovitskiy et al.[25]	83.58 ± 0.53	92.06 ± 0.12	87.56 ± 0.53	89.75 ± 0.33
Wang et al.[26]	94.50 ± 0.51	90.41 ± 0.51	92.34 ± 0.58	**95.01 ± 0.33**
M-MNet	93.19 ± 0.36	93.11 ± 0.16	93.19 ± 0.36	93.15 ± 0.16
FSO+M-MNet	**94.62 ± 0.15**	**94.07 ± 0.13**	**94.62 ± 0.15**	94.34 ± 0.12

5. Conclusion

This paper proposes a feature extraction method that combines the construction of gene mutation maps based on the RGB three-channel principle with feature selection optimization. Furthermore, we present an M-MNet classification model based on the inverted residual module and multi-head self-attention module. To overcome the issues of high noise and sparsity in gene mutation maps constructed based on the RGB three-channel principle, an FSO algorithm is applied for further feature extraction. At the same time, to design a classification model with better performance and stronger robustness, this paper proposes an M-MNet classification model based on the inverted residual module and multi-head self-attention module, trained using label smoothing regularization. The feature extraction method that combines the construction of gene mutation maps based on the RGB channel principle with feature selection optimization effectively solves the problem of high-dimensional mutation data being unsuitable for existing convolutional neural networks. Experimental results show that the overall performance of M-MNet is better than that of other existing classification models.

6. Acknowledgments

Not Applicable.

References

[1] F. Tomao, A. Papa, L. Rossi, M. Strudel, P. Vici, G. L. Russo, S. Tomao, Emerging role of cancer stem cells in the biology and treatment of ovarian cancer: basic knowledge and therapeutic possibilities for an innovative approach, Journal of Experimental & Clinical Cancer Research 32 (2013) 48–48.

[2] W. M. Linehan, C. J. Ricketts, The metabolic basis of kidney cancer, Seminars in Cancer Biology 23 (2012) 46.

[3] M. Meyerson, S. Gabriel, G. Getz, Advances in understanding cancer genomes through second-generation sequencing, Nature Reviews Genetics (2018).

[4] K. Sirvan, J. Peronne, B. Deepak, S. Salendra, L. F. Thomas, G. Kishore, V. Vinay, Sysmut: decoding the functional significance of rare somatic mutations in cancer, Briefings in Bioinformatics (2022) 4.

[5] J. Candia, E. Bayarsaikhan, M. Tandon, A. Budhu, X. W. Wang, The genomic landscape of mongolian hepatocellular carcinoma, Nature Communications 11 (2020).

[6] Abel, Gonzalez-Perez, Nuria, Lopez-Bigas, Functional impact bias reveals cancer drivers., Nucleic acids research (2012).

[7] L. Mularoni, R. Sabarinathan, J. Deu-Pons, A. Gonzalez-Perez, N. López-Bigas, Oncodrivefml: a general framework to identify coding and non-coding regions with cancer driver mutations, Genome Biology 17 (2016).

[8] H. Yi, Y. Juze, Q. Xinyi, C. Wei-Chung, L. Shu-Hsuan, H. Xing, Z. Liyuan, Y. Yaning, W. Qingbiao, L. a. Pengyuan, Driverml: a machine learning algorithm for identifying driver genes in cancer sequencing studies, Nuclc Acids Research (2019) e45–e45.

[9] B. Wei, K. Yang, J. Zhao, Y. Chang, Z. Ma, B. Dong, Y. Guo, J. Ma, Quantification of egfr mutations in primary and metastatic tumors in non-small cell lung cancer, Journal of Experimental & Clinical Cancer Research (2014).

[10] T. Zhao, C. Ren, Z. Zhang, H. Liu, Y. Zhang, Landscape of homologous recombination-related (hrr) genes mutations in colon cancer, Journal of Clinical Oncology 39 (2021) e15525–e15525.

[11] S. W. K. Ng, F. J. Rouhani, S. F. Brunner, N. Brzozowska, S. J. Aitken, M. Yang, F. Abascal, L. Moore, E. Nikitopoulou, L. a. Chappell, Convergent somatic mutations in metabolism genes in chronic liver disease, Nature 598 (2021) 473–478.

[12] W. Poole, K. Leinonen, I. Shmulevich, T. A. Knijnenburg, B. Bernard, Multiscale mutation clustering

algorithm identifies pan-cancer mutational clusters associated with pathway-level changes in gene expression, PLOS Computational Biology (2017).

[13] C. Tokheim, R. Bhattacharya, N. Niknafs, D. M. Gygax, R. Kim, M. C. Ryan, D. Masica, R. Karchin, Exome-scale discovery of hotspot mutation regions in human cancer using 3d protein structure, Cancer Research (2016) 3719.

[14] T. Wiegmann, I. Pacheco, F. Reikowski, J. Stettner, C. Qiu, M. Bouvier, M. Bertram, F. Faisal, O. Brummel, J. Libuda, Identification of the reversible skin layer on co, ACS catalysis 12 (2022) 3256–3268.

[15] M. F. B. Asad, M. N. A. Hallak, A. Sukari, Y. Baca, M. Nagasaka, Prognostic impact of xpo1 mutations in metastatic non-small cell lung cancer (nsclc), Journal of Clinical Oncology 39 (2021) e20533–e20533.

[16] Leiserson, D. M. Mark, Hsin-Ta, Vandin, Fabio, Raphael, J. Benjamin, Comet: a statistical approach to identify combinations of mutually exclusive alterations in cancer, Genome Biology 16 (2015) 1–20.

[17] Y. A. Kim, S. Madan, T. M. Przytycka, Wesme: Uncovering mutual exclusivity of cancer drivers and beyond, arXiv e-prints (2016).

[18] R. L. Stevens, Deep learning in cancer and infectious disease: Novel driver problems for future hpc architecture (2017).

[19] C. Quan, F. Liu, L. Qi, Y. Tie, Lrt-cluster: A new clustering algorithm based on likelihood ratio test to identify driving genes, Interdisciplinary Sciences: Computational Life Sciences 15 (2023) 217–230.

[20] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, A. Vaswani, Bottleneck transformers for visual recognition, 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021) 16514–16524.

[21] Q. Huang, D. Wang, Z. Lu, S. Zhou, J. Li, L. Liu, C. Chang, A novel image-to-knowledge inference approach for automatically diagnosing tumors, Expert Systems with Applications 229 (2023) 120450. URL: https://www.sciencedirect.com/science/article/pii/S0957417423009521. doi:https://doi.org/10.1016/j.eswa.2023.120450.

[22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1–9. doi:10.1109/CVPR.2015.7298594.

[23] D. Wang, Y. Wang, Y. Wang, L. Liu, J. Li, Q. Huang, Fusion of human cognitive knowledge and machine inference for breast cancer detection, in: 2023 International Conference on Advanced Robotics and Mechatronics (ICARM), 2023, pp. 179–184. doi:10.1109/ICARM58088.2023.10218759.

[24] M. Schlichtkrull, T. Kipf, P. Bloem, R. van den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: Extended Semantic Web Conference, 2017. URL: https://api.semanticscholar.org/CorpusID:5458500.

[25] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, ArXiv abs/2010.11929 (2020). URL: https://api.semanticscholar.org/CorpusID:225039882.

[26] D. Wang, Y. Wang, Y. Wang, L. Liu, J. Li, Q. Huang, Fully automated interpretable breast ultrasound assisted diagnosis system, in: 2023 International Conference on Advanced Robotics and Mechatronics (ICARM), 2023, pp. 173–178. doi:10.1109/ICARM58088.2023.10218807.