Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

CTransCNN: Combining transformer and CNN in multilabel medical image classification

Xin Wu [a], Yue Feng [a,*], Hong Xu [a,b], Zhuosheng Lin [a], Tao Chen [a], Shengke Li [a], Shihan Qiu [a], Qichao Liu [a], Yuangang Ma [a], Shuangsheng Zhang [c]

[a] *Faculty of Intelligent Manufacturing, Wuyi University, Jiangmen 529020, Guangdong, China*
[b] *Institute for Sustainable Industries and Liveable Cities, Victoria University, Melbourne 8001, Australia*
[c] *Jiangmen Central Hospital, Jiangmen 529030, Guangdong, China*

ARTICLE INFO

Keywords:
Multilabel image classification
Hybrid network
Convolutional neural networks
Transformer
Label correlation

ABSTRACT

Multilabel image classification aims to assign images to multiple possible labels. In this task, each image may be associated with multiple labels, making it more challenging than the single-label classification problems. For instance, convolutional neural networks (CNNs) have not met the performance requirement in utilizing statistical dependencies between labels in this study. Additionally, data imbalance is a common problem in machine learning that needs to be considered for multilabel medical image classification. Furthermore, the concatenation of a CNN and a transformer suffers from the disadvantage of lacking direct interaction and information exchange between the two models. To address these issues, we propose a novel hybrid deep learning model called CTransCNN. This model comprises three main components in both the CNN and transformer branches: a multilabel multihead attention enhanced feature module (MMAEF), a multibranch residual module (MBR), and an information interaction module (IIM). The MMAEF enables the exploration of implicit correlations between labels, the MBR facilitates model optimization, and the IIM enhances feature transmission and increases nonlinearity between the two branches to help accomplish the multilabel medical image classification task. We evaluated our approach using publicly available datasets, namely the ChestX-ray11 and NIH ChestX-ray14, along with our self-constructed traditional Chinese medicine tongue dataset (TCMTD). Extensive multilabel image classification experiments were conducted comparing our approach with excellent methods. The experimental results demonstrate that the framework we have developed exhibits strong competitiveness compared to previous research. Its robust generalization ability makes it applicable to other medical multilabel image classification tasks.

1. Introduction

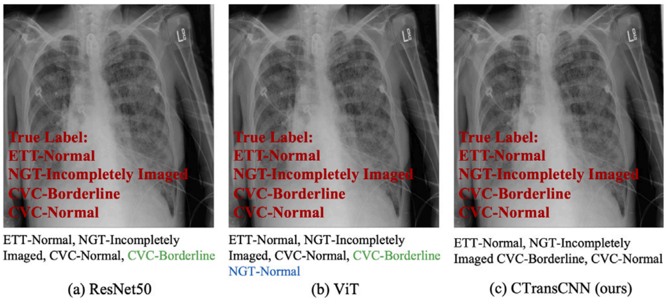
Multilabel image classification is a crucial task in which each data sample may be assigned multiple labels, rather than just a single label. It is very common in practical applications and can be applied to various scenarios, such as topic classification for article columns, medical diagnosis, image annotation, and recommendation systems.

In recent years, there have been significant advancements in multilabel image classification, largely attributed to deep learning techniques [1]. Specifically, CNNs have demonstrated remarkable performance [2]. Gong et al. [3] introduced a multilabel image classification method with a weighted approximate ranking loss. This method achieved good results on several datasets; however, it requires considerable training data and computing resources, and may not perform well for noisy labels and imbalanced datasets. Wei et al. [4] utilized a flexible method that takes an arbitrary number of object local region hypotheses as inputs to a shared CNN and fuses the predictions of each hypothesis with max pooling to obtain the final multilabel prediction. Wang et al. [5] designed the EfficientNet, which is composed of a feature extractor and a multilabel classifier. It can directly detect one or more fundus diseases in retinal fundus images for ODIR 2019[1] fundus images. At the same time, due to the correlation between labels, when a rare label appears, it is often accompanied by other labels with higher frequency [6]. Limited by convolutional kernels' representational capacity, CNN-based multilabel image classification approaches may not be able to fully exploit this statistical dependence, resulting in poor classification performance for rare labels. In addition, due to the limitation of receptive field size, CNN-based methods are

[*] Corresponding author.
E-mail address: J002443@wyu.edu.cn (Y. Feng).

[1] ODIR 2019: https://odir2019.grand-challenge.org/.

https://doi.org/10.1016/j.knosys.2023.111030
Received 12 July 2023; Received in revised form 31 August 2023; Accepted 21 September 2023
Available online 30 September 2023



(a) ResNet50 (b) ViT (c) CTransCNN (ours)

Fig. 1. Examples of recognition results of the CNN (ResNet50 [13]), the transformer (ViT) and our proposed CTransCNN. The true labels are in red font, the incorrectly identified labels are in green font, and the labels correctly predicted with smaller probability are in blue font.

usually unable to capture the long-range correlations between objects in the image [7].

As a result, some works have employed transformers to alleviate the above problems. Wang et al. [8] initially passed the image through the VGG16 [9] network and then utilized a spatial transformer (ST) to capture informative regions, followed by long short-term memory networks (LSTM) to model label correlations. Nie et al. [10] further enhanced this approach by replacing the ST module with an attentive transformer localizer module, which can flexibly integrate with LSTM and discover distinct semantic-aware regions in multilabel recognition. To better model complex and uncertain spatial label correlations, inspired by the remarkable success of vision transformer (ViT) [11] in image classification tasks, Chen et al. [12] proposed a plug-and-play module named the spatial and semantic transformer. They first extract holistic deep features using a CNN backbone and reshape the extracted features into sequences based on pixel positions.

The above methods have achieved effects, but there are still some challenges in research on multilabel medical image classification tasks. First, there may exist correlations between different anatomical structures and abnormalities in the images. For example, in multilabel CXR images, certain lung abnormalities may be related to cardiac anomalies, requiring consideration of the interdependencies among different labels in the classification task. Second, acquiring a sufficient amount of real medical images, especially for rare diseases, is difficult. This leads to label frequency imbalance, where some labels occur more frequently while others occur less frequently. Third, the distribution of lesion locations may exhibit a more widespread pattern throughout the entire image, and the features of different abnormalities may also have dispersed distributions. This means that there may be distinct local lesion features as well as scattered global features in the images.

In this paper, we propose a novel parallel hybrid framework named CTransCNN to alleviate the above problems. For the first problem, we introduce label embedding for self-attention operations. This approach captures the label correlations adaptively rather than relying on manually predefined label relationships. For the second problem, we employ cross-attention between image features and label features, allowing the model to weight the image features based on the importance of each label. This enables the model to pay more attention to the features of low-frequency labels, mitigating the impact of label imbalance and improving the classification accuracy for rare labels. For the third problem, we utilize a parallel structure that allows information interaction between CNN and transformer instead of a simple concatenation. CNN can provide richer inputs for the transformer by bottom-up feature extraction, while the transformer can guide the feature extraction of CNN through top-down attention mechanisms. This information interaction enhances the collaboration between the two components and improves the model's performance. At the same time, this interactive approach allows CNN, which excels at extracting local features, to better model the global features extracted by the transformer. To better demonstrate the superiority of our approach, we visualize the result of CNN (ResNet50 [13]), the transformer (ViT) and our proposed CTransCNN, as shown in Fig. 1.

Table 1
Abbreviations in the paper.

Abbreviation	Full name
CNNs	Convolutional neural networks
MMAEF	Multilabel multihead attention enhanced feature
MBR	Multibranch residual
IIM	Information interaction module
MSS	Multilabel semantic similarity
TCMTD	Traditional Chinese medicine tongue dataset
CXR	Chest x-ray
ViT	Vision transformer
C2T	The CNN branch to the transformer branch
T2C	The transformer branch to the CNN branch
MIML	Multi-instance multilabel
RNN	Recurrent neural network
ASL	Asymmetric loss
FFN	Position-wise feedforward network
BCE	Binary cross-entropy
LN	Layer normalization
BN	Batch normalization
TCM	Traditional Chinese medicine

The main contributions in this paper can be concluded as follows:

(1) Parallel Hybrid Architecture for Multilabel Medical Image Classification: We introduce a novel parallel hybrid architecture that combines both CNN and transformer. In this architecture, the CNN branch incorporates the MBR with inner and outer nested branches, while the transformer branch features the MMAEF with label embedding and the MSS block. Our goal is to effectively leverage these components to uncover the implicit correlations among labels and improve multilabel medical image classification.

(2) Cross-Branch Interaction via IIM: To enhance the model's nonlinearity and representation capability, we incorporate the IIM, namely C2T and T2C. These modules enable cross-branch communication and facilitate the exploration of implicit correlations between labels.

(3) Comprehensive Evaluation and Performance: We extensively evaluated our proposed framework, CTransCNN, on three distinct datasets: ChestX-ray11, NIH ChestX-ray14, and our in-house TCMTD. The evaluation demonstrated superior performance compared to existing models across all three datasets, highlighting the efficacy of our approach for multilabel medical image classification.

Table 1 lists all abbreviations in the paper.

2. Related work

In view of the three problems mentioned above, in this section, we provide a brief overview of previous research on multilabel image classification tasks, specifically focusing on label dependency, data imbalance and extensive lesion location.

2.1. Multilabel image classification methods on label dependency

The accuracy of the classifier can be affected by dependencies between different labels, where certain labels may only appear in the presence of other labels. Song et al. [14] proposed a deep multimodal CNN that combines CNN with MIML learning. It automatically generates instances for MIML by leveraging the structure of CNN, exploits label correlations by grouping them, and incorporates contextual information of label groups to generate multimodal instances. Allaouzi et al. [15] enhanced the accuracy and reliability of disease diagnosis by integrating a CNN model with convolutional filters that enable the detection of local patterns in images. This approach effectively improves the discrimination of features and labels in image analysis, leading to more precise and reliable disease diagnosis. Wang et al. [16] introduced a hybrid approach for addressing the challenge of multilabel

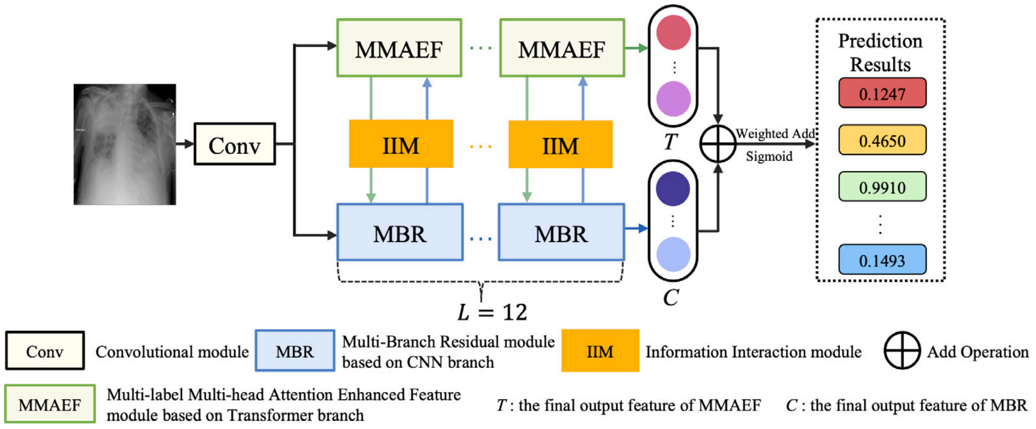

Conv
MMAEF
MMAEF
IIM
IIM
MBR
MBR
T
C
Weighted Add
Sigmoid
Prediction Results
0.1247
0.4650
0.9910
0.1493
L = 12
Conv Convolutional module
MBR Multi-Branch Residual module based on CNN branch
IIM Information Interaction module
Add Operation
MMAEF Multi-label Multi-head Attention Enhanced Feature module based on Transformer branch
T : the final output feature of MMAEF
C : the final output feature of MBR

Fig. 2. An overview diagram of the proposed CTransCNN framework, where the transformer branch employs the MMAEF block (green part), the CNN branch utilizes the MBR block (blue part), and the IIM incorporates C2T and T2C (orange part). Here, L represents the total number of stacks of the MMAEF and MBR modules. The horizontal colored arrows represent inter-layer information propagation, while the vertical colored arrows represent inter-branch information propagation.

image classification by integrating RNN and CNN models. By combining the strengths of RNN and CNN, it better utilizes the label dependencies within the image and facilitates the learning of joint image-label embeddings through end-to-end training.

The representational capacity of the convolutional kernel does impose constraints on the accuracy of multilabel image classification. Taslimi et al. [17] proposed a multilabel classification deep model based on the swin transformer backbone, which predicts each label using shared components across models. Recent visual transformer networks leverage self-attention mechanisms to extract pixel-level features and convey richer local semantic information. However, this approach still falls short of fully exploring global spatial dependencies. Some researchers utilized transformers to model complex dependencies between visual features and labels. Lanchantin et al. [18] designed the classification transformer framework for multilabel image classification. They trained a transformer encoder using label masks, which represent the state of a label as either positive, negative, or unknown during training, using a ternary encoding scheme. Zhu et al. [19] introduced a new method named the two-stream transformer to address the multilabel image classification problem. By leveraging attention mechanisms to learn the interaction between label semantics and high-level visual representations, the approach achieves accurate and robust alignment.

Certain researchers have identified that integrating label information as edge features into the model can enhance its perceptual capability for label correlation. Lee et al. [1] proposed a hybrid deep learning model based on CNN and graph neural networks to explore the implicit correlations among chest diseases and aid in multilabel CXR image classification tasks. It enhances the correlations among chest diseases by performing message passing and aggregation among the nodes.

2.2. Multilabel image classification methods on data imbalance

In traditional multilabel image classification, the handling of positive and negative samples is the same, so it cannot solve the problem of imbalanced positive and negative samples. However, Ridnik et al. [20] proposed an ASL function to solve the issue of imbalanced positive and negative samples in multilabel classification tasks. This method performs better in multilabel classification tasks because the hyperparameters of the ASL function can be dynamically adjusted to better handle the problem. Multilabel image classification requires a significant amount of annotated data, but obtaining and annotating data can often be costly. Yi et al. [21] designed MLSL-Net, a multilabel softmax network that addresses data imbalance and statistical label dependence in CNN-based multilabel classification of pulmonary nodules. MLSL-Net utilizes a strategy for extracting multiscale features and incorporates a multilabel softmax loss function. To address the issues of inconsistent target scales and imbalanced labels, Yan et al. [22] introduced the feature attention network, which incorporates a feature refinement network and a correlation learning network. It utilizes a top-down feature fusion mechanism to extract more important features, enabling it to learn the correlations between convolutional features and subsequently, the dependencies between labels.

2.3. Multilabel image classification methods on extensive lesion location

Medical images can contain multiple distinct lesion regions, which might manifest in various positions and areas within the image. Unlike the task of a single lesion location, multilabel medical image analysis requires a model capable of simultaneously identifying and labeling multiple lesion locations in an image. Zhou et al. [23] proposed a novel attention-augmented memory network model. They employed a categorical memory module to my contextual information of various label categories from the dataset to enhance features. They also designed a new channel relationship exploration module and a spatial relationship enhancement module to capture the interchannel relationships of features and the relationships between pixels in the feature maps. Liu et al. [24] leveraged a transformer decoder to query the presence of class labels and detect and aggregate the related features between labels in feature maps, which were ultimately utilized for binary classification. The effectiveness of the model was validated on five multilabel classification datasets and consistently outperformed all previous works.

3. Proposed method

The proposed approach for multilabel medical image classification consists of three main stages, as shown in Fig. 2. The first stage is to extract the initial features (e.g., edge and texture information) using the Conv module and then send two copies of them to the transformer branch and the CNN branch, respectively. In the second stage, the transformer branch adopts the label embedding and the MSS block of the MMAEF, while the CNN branch utilizes the MBR with nested inner and outer branches. The stacking of the MMAEF and MBR is equal to the number of layers in a vanilla transformer, denoted as $L = 12$. We believe that our model's number of layers is on par with the original architecture, which enhances structural reusability. Additionally, the widely recognized ViT model employs the transformer architecture for computer vision tasks and also utilizes a foundational version with 12 layers. Meanwhile, the IIM consists of the C2T and the T2C components to progressively fuse the feature maps in an interactive manner. Finally, after obtaining features T and features C from the two branches, we investigate three fusion methods for their classification: direct addition of the branch scores, weighted addition of the branch scores (with weight coefficients ranging from 0 to 1), and classification based on the concatenation of the final feature maps from the two branches. Through a series of experiments, it was found that setting the weight factor for weighted addition to 0.9 yields the best classification performance. In this section, we introduce each proposed module and compare the relevant loss function methods. For the training algorithm of CTransCNN, see Algorithm 1.

Algorithm 1: The training algorithm of CTransCNN.

```
Given An image dataset D=[$x_k$];
Use the Conv module to obtain the base features;
epoch is the number of iterations of training;
$b$ is the number of images in a batch;
$\mathcal{L}_{pareto}$ is a new loss function obtained by combining focal loss and
  ASL using pareto theory;
$n$ is the number of images in the dataset;
for $k \leftarrow 0$ to $epoch$ do
    for $l \leftarrow 0$ to $n$ by $b$ do
        In the batch instance $x = \{x_l^{l+b}\}$, the instance of CNN branch
          batch processing is $x_{CNN}$, and the instance of transformer
          branch batch processing is $x_{transformer}$ ;
        $x^i_{CNN} \Leftrightarrow x^i_{transformer}$, where $i$ represents layer $i$, $\Leftrightarrow$ represents
          the information interaction between MMAEF module and
          MBR module;
        $\mathcal{L}_{pareto}$, where $x_{final}$ represents the output of the fusion of the
          last layer of CNN and transformer branches;
        Perform backpropagation to update parameters;
    end
end
```

3.1. Label embedding and the MSS block of MMAEF

We employ a decoder-like approach, specifically the MMAEF, as shown in Fig. 3. Given an input image x, predict the presence of each class in a set of multilabel image data, such as a tongue image sample that can have one or more body constitution diagnosis outcomes. Assuming there are a total of C classes, we represent the corresponding label of x as $L = \{I_1, I_2, \dots, I_C\}$, where $I_i \in \{0,1\}, i = 1, \dots, C$, is a discrete binary indicator. If $I_i = 1$, it indicates that image x has the ith class label, otherwise $I_i = 0$. Using x as input, our model predicts the probability of the presence of each class, $p = [p_1, p_2, \dots, p_C]$, where $p_i \in [0,1], i = 1, \dots, C$.

For an input image x, it is first passed through a 7 × 7 convolution and a 3 × 3 max pooling to extract features. Then, in the transformer branch, a label embedding is used to query the MMAEF. However, most existing works primarily focus on regression from inputs to binary labels, while overlooking the relationship between visual features and the semantic vectors of labels. Specifically, we obtain the features extracted by the convolution as the key (K) and value (V) inputs and the label embeddings as the query $Q_i \in \mathbb{R}^{C \times d}$, with d denoting dimensionality, cross noting the desired K, V and Q. We use a transformer-like architecture, which includes a self-attention module, a cross-attention module, a MSS block (as shown in Fig. 4), and a FFN. When using the self-attention module, label embedding is the conversion of labels into vector representations so that the computer can better understand and process them. Compared to masked multihead attention, the conventional self-attention mechanism considers the contextual information of the entire sequence without placing particular emphasis on the order between positions. By incorporating label embeddings into the MMAEF framework, models can effectively and automatically capture the semantics of the labels and make more accurate predictions for multiple labels associated with an input sample. The specific formula is as follows:

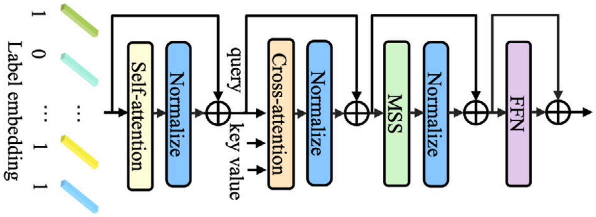


Fig. 3. Structure of the MMAEF. Label embedding is the introduction of multiple labels, self-learning using self-attention, combining image and semantic features by cross-attention, strengthening features using the MSS block, and finally getting the output probability.

$$\text{Self-attention: } Q_i^{(1)} = \text{MultiHead}\left(\tilde{Q}_{i-1}, \tilde{Q}_{i-1}, Q_{i-1}\right) \tag{1}$$

$$\text{Cross-attention: } Q_i^{(2)} = \text{MultiHead}\left(Q_i^{(1)}, K, V\right) \tag{2}$$

$$\text{MSS: } Q_i^{(3)} = \text{Concat}\left(Attention_1, \dots, Attention_h\right) W^O \tag{3}$$

$$Attention_j = \text{Softmax}\left(\text{Sigmoid}\left(\frac{Q_i^{(2)} K^T}{\sqrt{d_k}}\right)\right) V \tag{4}$$

$$Q_i = \text{FFN}(Q_i^{(3)}) \tag{5}$$

where Q_{i-1} indicates the MMAEF layer i update query from the output of the previous layer, W^O is a learnable parameter used for fusion, h represents the number of attention heads, $j = 1, \dots, h$, which we set to 6, and d_k represents the vector dimension of q and k.

$\text{MultiHead}(Q, K, V)$ and $Q_i = \text{FNN}(x)$ have the same decoder definition as the standard transformer [25]. We did not use masked multihead attention, but instead used self-attention, as autoregressive prediction is not required in multilabel image classification. In the MSS block, the embedding is replicated three times, creating separate copies for the Q, K, and V. The scalar product of the query vector and the transposed key vector is applied element-wise, and the resulting products are scaled by the square root of the key vector dimension. The sigmoid function is then used to map the variables to the range of 0 to 1, increasing the probability of label outputs and improving the performance of multilabel classification. Afterwards, normalization is performed using the softmax function, so that the sum of all elements is equal to 1. The resulting values are then used as weights to linearly combine the value vectors in V, returning the output of this weighted sum. Finally, this output is used as the final association score between labels, which is weighted to each label.

In multilabel medical image classification using cross-attention, the interplay between image feature representation and labels is commonly represented in the image and the labels are often modeled as a matrix product. This matrix product is computed through two distinct attention mechanisms, where one attention mechanism attends to regions in the image and the other attends to the interactions between the labels. In the case of multilabel medical classification, the results are processed by calculating the attention weights for each label using the sigmoid function, which is then multiplied by the value vector sequence and normalized using the softmax function to obtain the weight distribution for each position. By applying cross-attention between image features and label features, we can measure the image features based on the importance of each label.

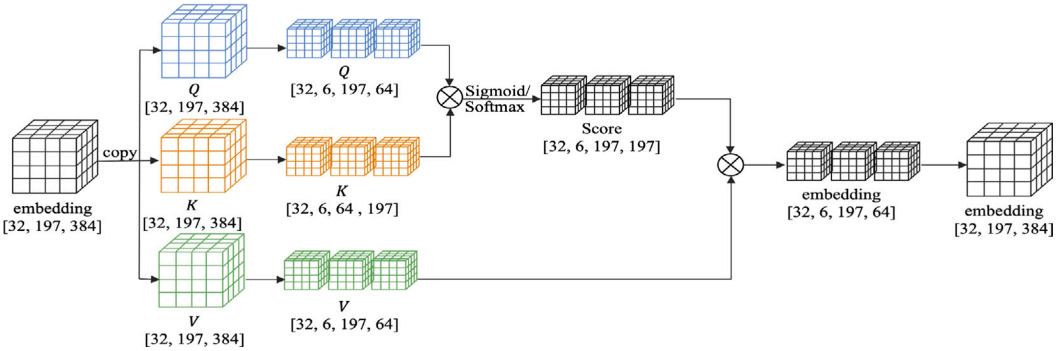


Fig. 4. Structure of the MSS block. The embedding is replicated in three copies of Q, K, and V. Firstly, Q and K are multiplied together to increase the label prediction probability using the sigmoid function, and then the output is normalized by softmax as the final correlation between the labels, weighted to each label.

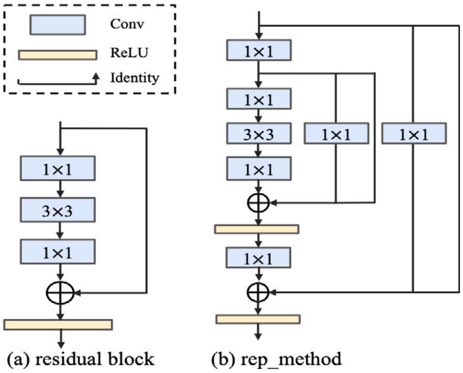


Fig. 5. Structure of the MBR: (a) residual block uses the standard residual connection; (b) rep_method increases the inner and outer nesting.

3.2. Multibranch residual module (MBR)

As shown in Fig. 5, the CNN branch of this paper adopts the nested structure of multibranch residual inner and outer branches, in which the resolution of the feature map decreases with the depth of the network. Following the definition in ResNet, we divide the entire CNN branch into four stages, each consisting of multiple convolutional blocks.

We adopt the basic bottleneck block of ResNet, which usually contains three convolutional layers. The first convolutional layer uses a smaller 1 × 1 downprojection convolution to reduce the dimensionality of the feature map. The second convolutional layer uses a larger 3 × 3 spatial convolution to extract features. And, the third convolutional layer again uses a smaller 1 × 1 upprojection convolution to reduce the dimensionality of the feature map. It also incorporates residual connections between the input and output, as illustrated in Fig. 5(a). The rep_method undergoes modification based on the residual block, wherein the primary 3 × 3 convolution is replaced by the residual block. This introduces an inner nesting process, accompanied by the addition of an inner branch, as shown in Fig. 5(b).

Compared with the residual block, rep_method uses an inner and outer nesting method. It allows the model to learn more detailed information. This design can reduce computational and parameter complexity while maintaining model performance. In CNN, the convolutional kernel slides and overlaps on the feature map, providing the possibility of retaining locally detailed features. Therefore, the CNN branch can continuously provide local feature details for the transformer branch through the C2T module (see Section 3.3 for details). The MBR improves the feature representation ability of the model, especially for multilabel classification tasks. It plays a crucial role in improving the model's ability to represent complex visual features and their relationships with multiple labels. By exploiting residual connections and combining information from different branches, the model can effectively learn and capture relevant features for accurate multilabel classification.

3.3. C2T and T2C in information interaction module (IIM)

For the CNN branch, it is a critical issue to map the features to the transformer. Similarly, for the transformer branch, it is also a very important issue to embed the patch embedding into the CNN. We realize that the feature dimensions of CNN and transformer are different. The CNN feature dimensions are $[B, C, H, W]$, where B represents the batch size, C represents a channel, H represents height, and W represents width. In contrast, the transformer feature dimensions are $[B, \_, C]$, where '_' represents the sum of the number of image patches and class tokens, usually $H{\times}W + 1$. To solve this problem, we propose the C2T and T2C approaches to gradually fuse feature maps in an interactive manner, as shown in Fig. 6.

The CNN branch to the transformer branch (C2T). We change the feature map dimensions using a 1 × 1 convolution. At the same time, we combine the feature information from different channels to enhance the expressive power of the features. The 1 × 1 convolution reduces the number of parameters in the model, thereby reducing computation and memory consumption. We use average pooling to downsample the feature map, reducing the spatial dimension while retaining the main information in the feature map. We use the GELU activation function, which allows for fast convergence and reduces training time, improving the efficiency of the model training. LN is used to regularize the features.

Specifically, let us consider $X \in \mathbb{R}^{H\times W\times C}$ representing a feature map of a convolutional MBR module, where C is the channel dimension, H and W represent the spatial dimensions (height and width). In other words, the output feature map has $H{\times}W$ pixel positions. Then, after the convolutional residual connection, we reshape these enhanced features along the spatial dimensions and obtain the flattened features:

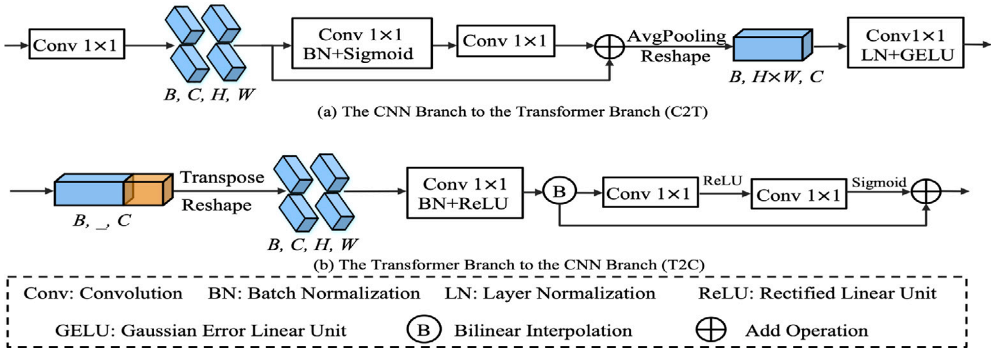
$$X_{cnn} = f_{conv}(X) + conv(X) \tag{6}$$

$$X'_{cnn} = X_{cnn} + X \tag{7}$$

$$X_{reshape} = \text{Reshape}\left(\text{AvgPool}\left(X'_{cnn}\right)\right) \tag{8}$$

$$X' = f_{conv}\left(X_{reshape}\right) \tag{9}$$

where $f_{conv}(\cdot)$ represents a 1 × 1 convolution operation with normalization and activation function. $conv(X)$ denotes a 1 × 1 convolution



(a) The CNN Branch to the Transformer Branch (C2T)

(b) The Transformer Branch to the CNN Branch (T2C)

Fig. 6. Structure of the IIM, which includes the C2T and T2C. Feature maps are collected from local convolution operators, while patch embeddings are aggregated by a global self-attentive mechanism. Therefore, C2T and T2C are applied in each block (except the first block) to fill the semantic gaps step by step.

operation. AvgPool refers to the average pooling operation. Reshape $(\cdot)$ represents the reshaping operator. $X_{reshape} \in \mathbb{R}^{(H \times W) \times C}$ denotes the flattened features.

The transformer branch to the CNN branch (T2C). The appropriate upsampling alignment space scale is used when going from the transformer branch to the CNN branch. Additionally, BN is used to regularize the features. The ReLU activation function, commonly used in convolution operations, is used. Upsampling the feature map using bilinear interpolation can improve the spatial resolution and thus capture more detailed information. Similar to C2T, we also use multiple 1 × 1 convolutions to optimize features for information exchange. However, after the 1 × 1 convolution, we cross both ReLU and sigmoid activation functions to improve the nonlinear fitting ability. Finally, residual connections are added between the output of bilinear interpolation and the output after a series of operations to preserve the importance of the features and improve the representation capability of the model.

In particular, let us consider $X' \in \mathbb{R}^{(H \times W+1) \times C}$ as the feature map of an MMAEF module, where 1 represents the class token. First, we remove the class token through a simple operation to obtain X_{non_cls}. Then, the sequence is restored back to a 2D feature map. Finally, we preserve more image detail information through convolutional operations and bilinear interpolation. The specific expression is as follows:

$$X_{non\_cls} = X'\left[:, 1:\right] \tag{10}$$

$$X_{restore} = \mathrm{Restore}\left(X_{non\_cls}\right) \tag{11}$$

$$X_{trans} = f_{conv}\left(X_{non\_cls}\right) \tag{12}$$

$$X'_{trans} = f_{conv\_s}(f_{conv\_r}\left(\mathrm{B}X_{trans}\right)) \tag{13}$$

$$X = f_{conv}\left(X'_{trans}\right) + \mathrm{B}\left(X_{trans}\right) \tag{14}$$

where X_{non_cls} represents the tensor without the class token, and Restore $(\cdot)$ represents the inverse operation of Reshape $(\cdot)$; f_{conv_r} represents a 1 × 1 convolution operation followed by the ReLU function, and f_{conv_s} represents a 1 × 1 convolution operation followed by the sigmoid function; B $(\cdot)$ represents the bilinear interpolation operation.

3.4. Customized multilabel loss function

The commonly used loss function for multilabel image classification, BCE loss [26], is widely applied in the early stages of research and development in this field. Focal loss [27] is an improvement over BCE loss that mainly focuses on reducing the weight of easily classified samples and increasing the weight of hard-to-classify samples, making the model more attentive to difficult samples. Compared to BCE loss, focal loss performs better in handling class imbalance problems and paying more attention to hard-to-classify samples. It is suitable for multiclassification problems and converges faster. Recently, considering the problem of label imbalance, Ridnik et al. [20] proposed an ASL, which can assign different punishments to misclassified labels of different classes according to the actual situation, making the model more focused on classes with fewer samples.

The proposed CTransCNN hybrid network outputs a logarithm for each of the C labels, denoted as $h_i, i = 1, \ldots, C$, which is then independently activated through the sigmoid function $\sigma\left(h_i\right)$. The total classification loss, L_{total}, is obtained by summing up the losses of the C labels.

$$L_{total} = \sum_{i=1}^{C} L\left(\sigma\left(h_i\right), y_i\right) \tag{15}$$

Focal loss is obtained by setting L_+ and L_-, with the specific formula as follows:

$$\begin{cases} L_+ = (1-p)^{\gamma} log\,(p) \\ L_- = p^{\gamma} log\,(1-p) \end{cases} \tag{16}$$

where $p = \sigma\,(h)$ is the output probability of the network, and γ is the focusing parameter. When $\gamma = 0$, the formula reduces to BCE loss.

Shifting the loss function by a factor m, the ASL function is defined as follows:

$$p_m = max\,(p-m, 0) \tag{17}$$

$$ASL = \begin{cases} L_+ = (1-p)^{\gamma_+} log\,(p) \\ L_- = \left(p_m\right)^{\gamma_-} log\left(1-p_m\right) \end{cases} \tag{18}$$

where m is the shifting factor of the loss function, and γ_+ and γ_- denote the positive and negative focusing parameters, respectively.

A Pareto optimal solution is one in which “none of the objectives can be improved without sacrificing at least one of the other objectives”. Pareto optimal solutions emphasize trade-offs and balances in multi-objective optimization to avoid improving one objective while weakening others. Inspired by this, we combine Focal loss and ASL and use Pareto optimization theory to balance the two loss functions. We put the two losses into a list, calculate the Pareto front and update the loss weights.

4. Experiments

4.1. Datasets

Three datasets were used to evaluate our proposed CTransCNN method: two publicly available multilabel CXR datasets, and a self-built multilabel tongue image dataset for TCM constitution classification.

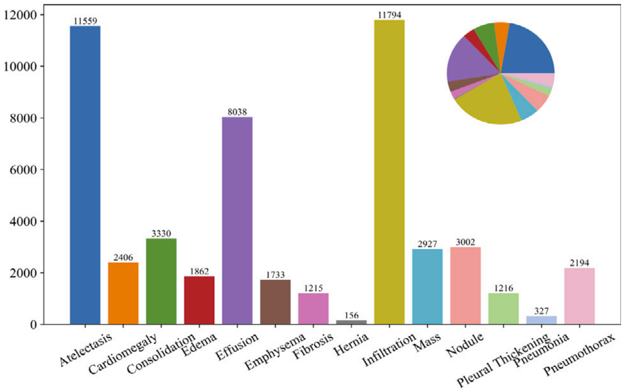


Fig. 7. Comparison of the number of cases for each disease in the NIH ChestX-ray14 dataset.

Table 2
The number of the ChestX-ray11 images for each disease. There are serious data imbalances, for example, there are many more CVC-Normal images than ETT-Abnormal images.

Disease	Number of images	Percentage
ETT-Abnormal	79	0.16%
ETT-Borderline	1138	2.25%
ETT-Normal	7240	14.30%
NGT-Abnormal	279	0.56%
NGT-Borderline	529	1.05%
NGT-Incompletely Imaged	2748	5.43%
NGT-Normal	4797	9.48%
CVC-Abnormal	3195	6.32%
CVC-Borderline	8460	16.71%
CVC-Normal	21 324	42.10%
Swan Ganz Catheter Present	830	1.64%

The Catheter and Line Position Challenge on Kaggle[2] is a competition that involves classifying 40 000 images to detect misplaced catheters. Table 2 shows the number of ChestX-ray11 images for each disease. In this study, 30 083 CXR image training data were used for multilabel sample classification, which was named ChestX-ray11. There are a total of 11 different types of catheter placement, including ETT-Abnormal, ETT-Borderline, ETT-Normal, NGT-Abnormal, NGT-Borderline, NGT-Incompletely Imaged, NGT-Normal, CVC-Abnormal, CVC-Borderline, CVC-Normal, and Swan Ganz Catheter Present. Each image may have one or more types of catheter placement. The images are of varying sizes and are gray-scale.

The NIH ChestX-ray14 dataset,[3] which is an extension of the ChestX-ray8 [28], includes 112 120 frontal X-ray images from 30 805 unique patients with annotations for 14 common diseases. The images have a size of 1024 × 1024 pixels and are from special patient populations. The 14 pathologies in NIH ChestX-ray14 are Atelectasis, Cardiomegaly, Consolidation, Edema, Effusion, Emphysema, Fibrosis, Hernia, Infiltration, Mass, Nodule, Pleural Thickening, Pneumonia, and Pneumothorax. Fig. 7 shows the number of NIH ChestX-ray14 images for each disease.

The TCMTD is a multilabel classification task for 9 different TCM pathologies, conditions viz. 'Qixu' (qi deficiency), 'Qiyu' (qi stagnation), 'Shire' (damp heat), 'Tanshi' (phlegm damp), 'Tebing' (idiosyncratic), 'Xueyu' (blood stagnation), 'Yinxu' (yin deficiency), 'Pinghe' (balanced), and 'Yangxu' (yang deficiency), which is a multilabel classification task. The balanced constitution is 'Pinghe' and the rest are the imbalanced constitutions. The TCMTD involves 1050 student volunteers, of which 1019 images were usable. Some images were discarded due to poor quality resulting in unclear images, machine-lagging image capture failure, and the presence of tongue studs in the images. The entire study followed the requirement of Human Research Ethics (approval number: [2019] 18). Tongue images were captured with professional equipment in a closed environment, and multilabel constitutional labels were annotated by clinical TCM experts. The dataset includes 1019 tongue images from volunteers, with 7 types of patterns representing 9 different TCM constitutions (patterns of disharmonies) excluding 'Pinghe' and 'Yangxu'. Each image is labeled as one or more pathological conditions. The original dataset consists of a size of 1716 × 2574 pixels. Each image is labeled with $I = \{I_1, I_2, \dots, I_C\}$, and C is 7 in the tongue dataset. Each element of I is set as 0 for absence and 1 for presence, as shown in Table 3.

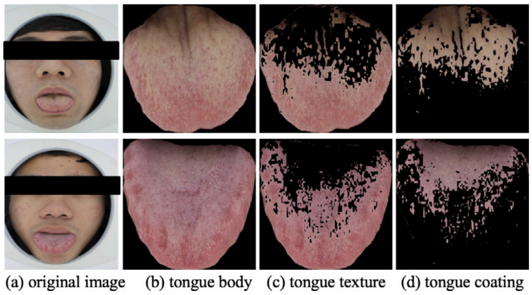

[redacted]
[redacted]
(a) original image (b) tongue body (c) tongue texture (d) tongue coating

Fig. 8. The preprocessing of tongue image analysis includes converting the raw tongue image into a tongue body image, followed by segmentation into tongue coating and tongue body images.

Table 3
Detail annotation of a sample of the TCMTD.

Qixu	Qiyu	Shire	Tanshi	Tebing	Xueyu	Yinxu
1	0	0	1	0	0	0
0	0	0	1	0	1	1
0	1	1	0	0	1	0
1	0	0	0	1	1	0
1	0	0	0	0	0	0
0	0	1	0	0	1	1

'Pinghe' and 'Yangxu' conditions among the nine TCM constitutions are not displayed due to sample collection limitations.

4.2. Image preprocessing

Because the ChestX-ray11 and NIH ChestX-ray14 are public datasets, simple label processing and size normalization are sufficient. However, the TCMTD is an undisclosed and variable-sized dataset, and the coating and body of the tongue are intertwined. Additionally, changes in the body and coating of the tongue are closely related to the body's organs, qi and blood, and the severity of pathogenic factors. Therefore, it is necessary to separate the tongue body and coating. First, based on the characteristics of tongue images, the multilayer edge attention network [29], which introduces attention mechanisms and fuses edge information, is used to segment the tongue from the background image. Second, after tongue segmentation, the tongue coating and body are separated to more accurately identify tongue and coating colors and extract tongue and coating features. The original RGB image is first converted to the Lab color space, and the a-channel image is enhanced and corrected based on color block differences. The K-means clustering algorithm is then used to separate the tongue coating and body. Separating the tongue coating and body facilitates network feature extraction, and a multilabel learning algorithm is used for classification. The preprocessing results of the tongue image analysis are shown in Fig. 8.

[2] ChestX-ray11: kaggle.com/competitions/ranzcr-clip-catheter-line-classification/data.

[3] ChestX-ray14: nihcc.app.box.com/v/ChestXray-NIHCC.

4.3. Evaluation criteria

In this study, we utilized seven widely recognized evaluation metrics for multilabel classification to assess the performance of different models. These metrics include:

- Mean Average Precision (mAP) across all categories.
- Overall Precision (OP).
- Overall Recall (OR).
- Overall F1-measure (OF1).
- Per-class Precision (CP).
- Per-class Recall (CR).
- Per-class F1-measure (CF1).

Furthermore, we employed receiver operating characteristic (ROC) curves and calculated the area under the curve (AUC) to compare the overall classification performance of each model on both the ChestX-ray11 and NIH ChestX-ray14 datasets. This approach provides a visually intuitive evaluation the model's performance. Specifically, we used p-values to measure the degree of difference between the observed data and hypotheses, as described in Section 4.5.

$$\mathrm{AP_i} = \frac{1}{|G_i|}\sum_{k=1}^{n} P_k \times rel_k \tag{19}$$

$$\mathrm{mAP} = \frac{1}{|\mathcal{Y}|}\sum_{i=1}^{|\mathcal{Y}|} \mathrm{AP}_i \tag{20}$$

where $|G_i|$ represents the number of samples in the ith category, P_k represents the precision of the top k predictions, rel_k represents whether the kth prediction is the true label of the sample (1 if yes, 0 if no), n represents the total number of predictions, y represents the set of all labels, and $\mathrm{AP_i}$ represents the average precision for the ith label.

$$\mathrm{CP} = \frac{1}{K}\sum_{i=1}^{K}\frac{TP_i}{TP_i + FP_i} \tag{21}$$

$$\mathrm{CR} = \frac{1}{K}\sum_{i=1}^{K}\frac{TP_i}{TP_i + FN_i} \tag{22}$$

$$\mathrm{CF1} = \frac{1}{K}\sum_{i=1}^{K}\frac{2\times TP_i}{2\times TP_i + FP_i + FN_i} \tag{23}$$

$$\mathrm{OR} = \frac{TP}{TP + FN} \tag{24}$$

$$\mathrm{OP} = \frac{TP}{TP + FP} \tag{25}$$

$$\mathrm{OF1} = \frac{2\times TP}{2\times TP + FP + FN} \tag{26}$$

where K represents the total number of categories, TP_i represents the number of true positives for ith category, FP_i represents the number of false positives for ith category, and FN_i represents the number of false negatives for ith category.

4.4. Implementation details

To ensure a fair comparison, we normalized the size of the input images to 224 × 224 for all experiments during the training, validation, and testing stages. We trained the entire network using the AdamW [30] algorithm and a cosine annealing strategy, with an initial learning rate of 1e-3, a momentum of 0.5 and 0.999, a single GPU batch size of 32, and 8 threads per single GPU. During the training process, we used data augmentation techniques such as random horizontal flipping, random scaling cropping, and RandAugment with a probability of 0.5. During the testing phase, only scaling was performed. In particular, we partitioned all three datasets into three subsets, including three subgroups (70% training, 20% validation, and 10% testing). Additionally, all experiments were implemented using Python 3.9 and PyTorch 1.11.0 in the CUDA 11.4 universal computing framework, running on an Ubuntu 20.04.2 operating system with one Nvidia A5000 GPU with 24 GB memory. We implemented our models using the open source computer vision library MMCV[4], which was developed by OpenMMLab. We would like to express our gratitude to the developers for their valuable contributions to the research community.

4.5. Comparative results on the ChestX-ray11 and NIH ChestX-ray14 datasets

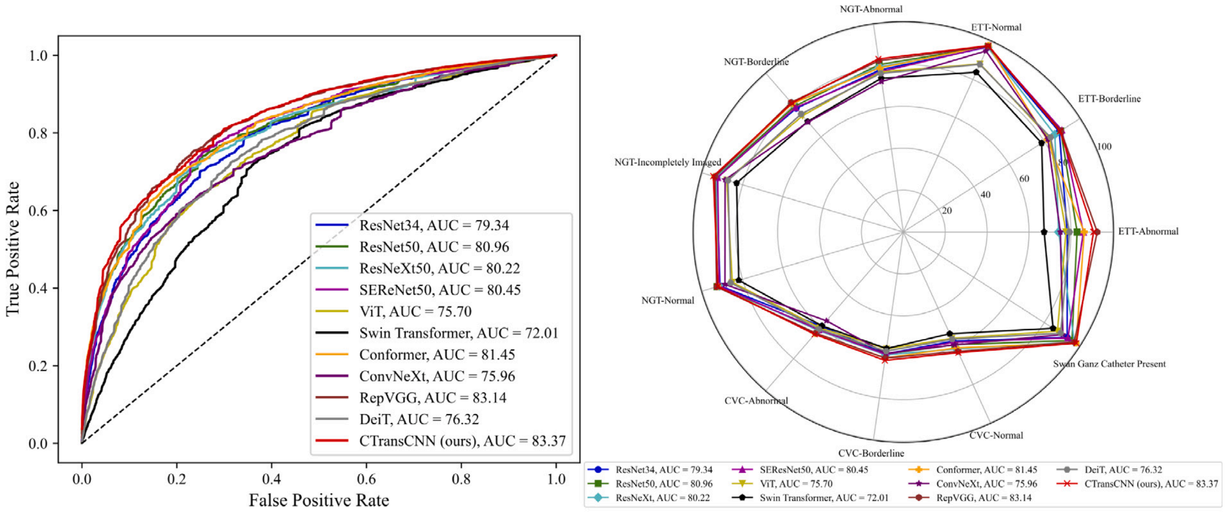
To ensure a fair comparison, we compute the AUC scores for each category and the average AUC score for all diseases using the aforementioned parameter settings and classic methods. Tables 4 and 5 present the experimental results on the ChestX-ray11 and NIH ChestX-ray14 datasets, respectively. The CTransCNN is compared with 10 excellent medical image classification networks, achieving the best performance in multilabel classification. Furthermore, we employed a paired t-test to evaluate the statistical significance of the performance differences between our proposed model and the models proposed by other authors. Based on the p-values, it can be concluded that there are statistically significant differences in performance across the models for this particular task.

As shown in Table 4, for the ChestX-ray11 dataset, the CTransCNN achieved the highest average AUC score (83.37%) compared to the other models. In terms of individual diseases, CVC-Normal had the lowest average AUC score (58.81%), while ETT-Normal achieved the highest average AUC score (94.23%). ResNet34, ResNet50, ResNeXt [31], and SEResNet50 [32] exhibited good average AUC values ranging from 79.34% to 80.45%. These models consistently performed well across multiple labels, making them reliable choices for multilabel image classification tasks. Conformer and RepVGG showed relatively higher average AUC values of 81.45% and 83.14%, respectively. These models demonstrated excellent performance on specific labels, with RepVGG [35] achieving the highest AUC score (92.58%) on ETT-Abnormal. ViT, Swin transformer, ConvNeXt, and DeiT [36] had relatively lower average AUC values ranging from 72.01% to 76.32%. Based on the average AUC values, RepVGG and CTransCNN appear to be the top-performing models, with CTransCNN outperforming RepVGG by a margin of 0.23% in terms of the average AUC score.

As shown in Table 5, for the NIH ChestX-ray14 dataset, based on the average AUC scores, our CTransCNN model performed well across most diseases and overall, with an average AUC score of 78.47%. Other models that performed relatively well include ConvNeXt (76.73%), Rep-VGG (76.53%), and Conformer (76.39%). The ViT model had the lowest average AUC score, at 54.00%. Among the different diseases, Pneumonia had the lowest average AUC score (57.58%), while Cardiomegaly had the highest average AUC score (82.58%), followed by Edema (82.51%). Through the above analysis, it can be observed that models with different architectures have varying performances across different disease classification tasks. For example, models such as Conformer, ConvNeXt, and RepVGG performed well in terms of average AUC scores, while transformer-based models such as ViT and DeiT had relatively lower scores. In terms of the average AUC value, ConvNeXt appears to be the top-performing model along with CTransCNN, but our proposed CTransCNN model outperformed ConvNeXt by a margin of 1.74% in the average AUC score.

To facilitate comparison with other algorithms and to showcase the network's overall performance, Figs. 9 and 10 illustrate the classification performance of the compared methods, along with the AUC for each disease. These visualizations pertain to the ChestX-ray11 and NIH ChestX-ray14 datasets, respectively. In Fig. 9(a) for the ChestX-ray11 dataset, the ROC curve of our method is closer to the top-left corner, indicating superior performance compared to other methods.

[4] github.com/open-mmlab/mmcv.



a) Comparing the ROC curves of different network models

b) Compare the AUC values of various diseases

Fig. 9. Comparison of the average AUC scores (%) of CTransCNN and 10 common networks on the ChestX-ray11 dataset. (a) Compare the ROC curves of different network models; (b) Compare the diagnostic model performance of 11 diseases.

Table 4
Comparison of the classification performance of our different models on the ChestX-ray11 dataset. The best results are shown in bold.

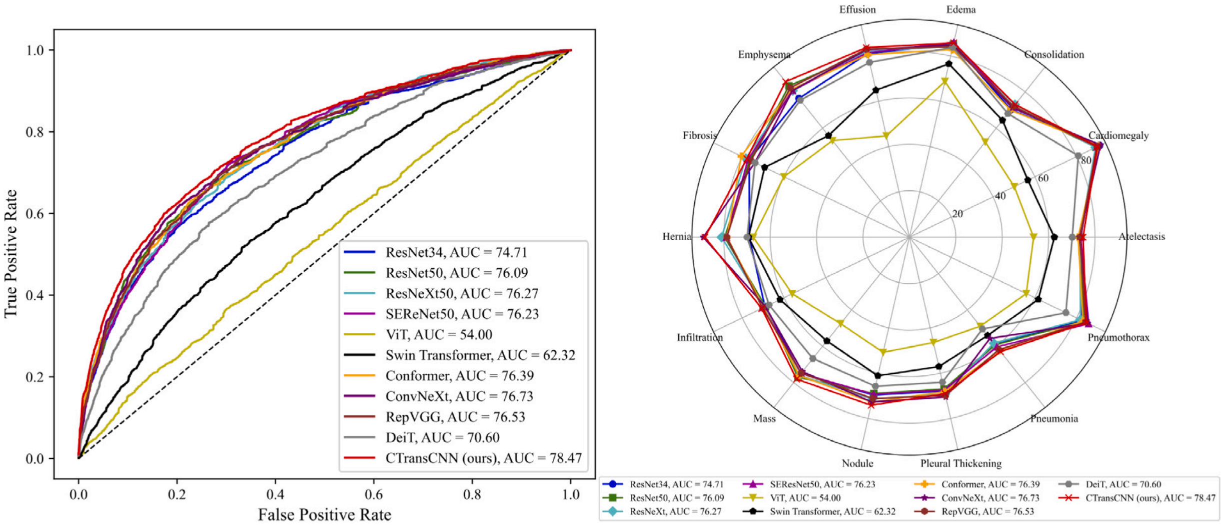
AUC score (%)	ResNet34 [13]	ResNet50 [13]	ResNeXt [31]	SEResNet50 [32]	ViT [11]	Swin transformer [17]	Conformer [33]	ConvNeXt [34]	RepVGG [35]	DeiT [36]	CTransCNN (ours)	Mean
ETT-Abnormal	78.26	82.97	74.16	85.95	77.68	67.23	86.39	74.96	**92.58**	79.36	90.82	80.95
ETT-Borderline	88.51	89.33	85.96	**89.96**	83.02	78.49	81.63	81.99	89.39	83.56	88.93	85.53
ETT-Normal	96.96	97.37	96.95	97.29	88.18	83.84	97.78	94.96	97.37	87.95	**97.80**	94.23
NGT-Abnormal	78.15	80.69	79.63	77.26	77.18	74.09	79.11	72.54	82.52	76.48	**83.42**	78.28
NGT-Borderline	78.10	79.35	81.14	78.58	73.41	69.74	81.02	69.40	**81.83**	74.51	81.32	77.13
NGT-Incompletely Imaged	92.94	93.09	93.2	92.37	87.88	82.97	93.87	88.66	93.67	87.25	**94.53**	90.95
NGT-Normal	91.19	92.72	91.57	91.55	85.09	81.86	92.36	88.74	**92.92**	85.93	92.24	89.66
CVC-Abnormal	59.64	61.26	62.11	61.66	60.60	59.29	64.14	56.05	63.84	61.44	**64.59**	61.33
CVC-Borderline	58.80	58.89	59.26	58.99	56.36	56.18	60.79	58.57	60.79	57.44	**61.90**	58.91
CVC-Normal	57.20	59.14	61.32	57.89	55.80	53.38	60.78	59.19	62.64	56.23	**63.32**	58.81
Swan Ganz Catheter Present	92.95	95.72	97.11	93.44	87.48	84.99	98.07	90.51	97.01	89.31	**98.19**	93.17
Mean	79.34	80.96	80.22	80.45	75.70	72.01	81.45	75.96	83.14	76.32	**83.37**	–
p-value*	.0016	.0034	.0235	.0008	.0000	.0000	.0130	.0001	.2208	.0000	–	–

* The p-values are calculated from the AUC comparison between the CTransCNN and the other 10 models, and a p-value < 0.05 is considered statistically significant.

Table 5
Comparison of the classification performance of our different models on the NIH ChestX-ray14 dataset. The best results are shown in bold.

AUC score (%)	ResNet34 [13]	ResNet50 [13]	ResNeXt [31]	SEResNet50 [32]	ViT [11]	Swin transformer [17]	Conformer [33]	ConvNeXt [34]	RepVGG [35]	DeiT [36]	CTransCNN (ours)	Mean
Atelectasis	73.35	73.87	73.51	73.81	53.53	62.48	72.68	73.31	73.57	70.19	**74.84**	70.47
Cardiomegaly	91.00	89.18	89.03	89.72	50.18	56.67	90.69	**91.42**	89.65	80.81	89.95	82.58
Consolidation	70.19	71.72	72.98	71.73	52.43	64.33	69.96	71.19	72.07	68.06	**73.08**	68.89
Edema	85.40	85.08	84.32	**86.02**	68.78	76.52	82.82	84.89	84.09	83.85	85.80	82.51
Effusion	81.02	81.79	82.09	81.40	44.74	64.95	80.62	82.77	83.06	77.17	**83.72**	76.67
Emphysema	76.39	83.29	80.63	80.88	53.19	55.94	81.45	80.79	82.28	75.45	**85.63**	76.00
Fibrosis	76.66	76.37	78.26	77.18	60.02	69.27	**80.43**	73.94	75.82	73.84	77.79	74.51
Hernia	69.00	79.62	80.96	78.88	67.29	69.35	78.62	**88.84**	78.72	69.99	88.14	77.22
Infiltration	69.12	68.85	69.24	69.78	56.07	61.89	69.73	68.87	70.25	67.16	**70.65**	67.42
Mass	74.96	77.04	76.07	74.92	47.56	57.07	76.14	73.97	74.82	66.77	**78.06**	70.68
Nodule	69.65	68.94	71.36	69.34	50.83	61.15	72.81	72.73	71.17	65.79	**74.17**	68.00
Pleural Thickening	67.83	66.99	69.23	67.43	46.46	57.09	68.13	**70.51**	69.87	63.99	69.01	65.10
Pneumonia	59.02	59.21	58.32	60.39	48.97	54.07	62.31	55.68	61.87	50.50	**62.99**	57.58
Pneumothorax	82.39	83.32	81.81	85.75	55.90	61.65	83.05	**85.34**	84.21	74.86	84.68	78.46
Mean	74.71	76.09	76.27	76.23	54.00	62.32	76.39	76.73	76.53	70.60	**78.47**	–
p-value*	.0077	.0007	.0012	.0039	.0001	.0001	.0072	.0123	.006	.0001	–	–

* The p-values are calculated from the AUC comparison between the CTransCNN and the other 10 models, and a p-value < 0.05 is considered statistically significant.



a) Comparing the ROC curves of different network models

b) Compare the AUC values of various diseases

Fig. 10. Comparison of the average AUC scores (%) of CTransCNN and 10 common networks on the NIH ChestX-ray14 dataset. (a) Compare the ROC curves of different network models; (b) Compare the diagnostic model performance of 14 diseases.

Table 6
Comparison of the classification performance (mean±standard deviation) of our different models on the TCMTD. The best results are shown in bold.

Network	AUC (%)	mAP (%)	CP (%)	CR (%)	CF1 (%)	OP (%)	OR (%)	OF1 (%)
ResNet34 [13]	78.33 ± 3.73	61.67 ± 5.36	54.70 ± 2.70	58.24 ± 3.31	56.41 ± 2.87	76.74 ± 1.48	85.83 ± 1.41	81.03 ± 1.36
ResNet50 [13]	82.66 ± 1.46	64.60 ± 2.96	58.67 ± 2.97	61.06 ± 2.51	59.81 ± 2.36	78.25 ± 1.19	86.64 ± 1.79	82.23 ± 1.35
ResNeXt [31]	82.90 ± 0.77	65.20 ± 0.98	59.88 ± 0.70	61.78 ± 0.77	60.81 ± 0.60	79.43 ± 0.43	86.85 ± 0.86	82.97 ± 0.35
SEResNet 50 [32]	82.87 ± 1.93	65.60 ± 2.37	58.29 ± 2.63	60.76 ± 2.37	59.48 ± 2.35	79.01 ± 1.07	87.43 ± 1.10	83.00 ± 0.90
ViT [11]	63.83 ± 4.73	47.47 ± 2.45	40.91 ± 1.55	**69.25 ± 1.84**	51.41 ± 1.49	58.31 ± 1.79	**95.57 ± 1.16**	72.40 ± 1.14
Swin transformer [17]	54.22 ± 2.54	43.59 ± 1.25	41.84 ± 4.89	60.05 ± 2.92	49.10 ± 2.97	62.99 ± 1.69	92.67 ± 1.26	74.97 ± 0.81
Conformer [33]	82.05 ± 1.04	63.47 ± 0.76	58.59 ± 2.31	62.15 ± 0.87	60.29 ± 1.27	78.15 ± 1.22	87.43 ± 1.12	82.53 ± 1.15
ConvNeXt [34]	72.34 ± 2.67	58.82 ± 3.68	45.91 ± 2.97	59.97 ± 1.46	51.94 ± 1.83	66.39 ± 3.30	84.55 ± 2.64	74.33 ± 2.52
RepVGG [35]	79.94 ± 2.56	63.43 ± 1.69	58.11 ± 2.59	61.87 ± 1.27	59.91 ± 1.83	78.40 ± 1.27	87.13 ± 1.34	82.53 ± 1.28
DeiT [36]	65.09 ± 2.34	50.49 ± 2.31	43.76 ± 0.99	67.05 ± 2.21	52.95 ± 1.22	61.05 ± 1.68	94.32 ± 1.44	74.09 ± 0.84
CTransCNN(ours)	**84.56 ± 1.16**	**67.67 ± 1.48**	**63.31 ± 4.55**	65.42 ± 2.44	**64.32 ± 3.46**	**79.51 ± 1.05**	89.31 ± 0.44	**84.12 ± 0.67**

Fig. 9(b) shows the AUC curves for each disease, where the red curve of our method is closer to the outermost arc, further confirming its effectiveness. For the NIH ChestX-ray14 dataset, Fig. 10 is the same.

In summary, across the ChestX-ray11 and NIH ChestX-ray14 datasets, the CTransCNN achieved the highest average AUC values of 83.37% and 78.47%, respectively, outperforming all other models. This indicates its strong overall performance on different labels, which can be attributed to the incorporation of the MMAEF module. This module facilitates the exploration of implicit correlations between diseases by utilizing a set of label embeddings and the MMS block. Additionally, the introduction of MBR optimization, C2T, and T2C enhances the model's representation capacity, proving crucial in addressing the challenges of imbalanced multilabel image classification tasks.

4.6. *Comparative results on the TCMTD*

Table 6 presents a quantitative study of the performance of CTransCNN and ten excellent classification networks on the TCMTD. All evaluation metrics in Table 6 are the average values and standard deviations of five experiments.

Quantitative analysis: According to Table 6, the CTransCNN outperforms the other models in the multilabel image classification task. Specifically, CTransCNN achieves the highest scores in both AUC and mAP metrics, with values of 84.56% and 67.67%, respectively. In addition, it also performs well in CP, CF1, OP, and OF1 metrics, surpassing the other models. These metrics indicate that CTransCNN developed by our team can more accurately predict the presence or absence of each label in multilabel image classification tasks. ResNet50 outperforms ResNet34 in all metrics, suggesting that the utilization of deeper residual blocks enhances the model's ability to capture image features. This improvement effectively addresses the vanishing gradient problem, leading to more effective training of the model. ViT, Swin transformer, and DeiT perform relatively poorly in multilabel image classification tasks. For small datasets or multilabel classification tasks, these models perform worse than ResNet models. RepVGG performs relatively well in some evaluation metrics, surpassing commonly used image classification models such as ResNet34 and ResNet50. Compared with ResNet34, ResNet50, ResNeXt50, SEResNet50, ViT, Swin transformer, Conformer, ConvNeXt, RepVGG, and DeiT, the CTransCNN improves the main evaluation metric AUC by 6.23%, 1.90%, 1.66%, 1.69%, 20.73%, 30.34%, 2.51%, 12.22%, 4.62%, and 19.47%, respectively. These experimental results demonstrate that the CTransCNN can effectively perform multilabel image classification and outperforms other comparative methods.

Qualitative analysis: We generated their heatmaps using Grad-CAM++, eigen_smooth for removing a large amount of noise, and aug_smooth for testing time augmentation, as shown in Fig. 11. Networks such as ResNet34, ResNet50, SEResNet50, and RepVGG pay more attention to local edge information. The heat map weight distribution of models like ViT, Swin transformer, and DeiT appears relatively uniform. This indicates their strong global cognitive capability, with global information spanning across the entire image. Although Conformer pays attention to both local and global information, the heatmaps of the model are not similar in different runs, indicating

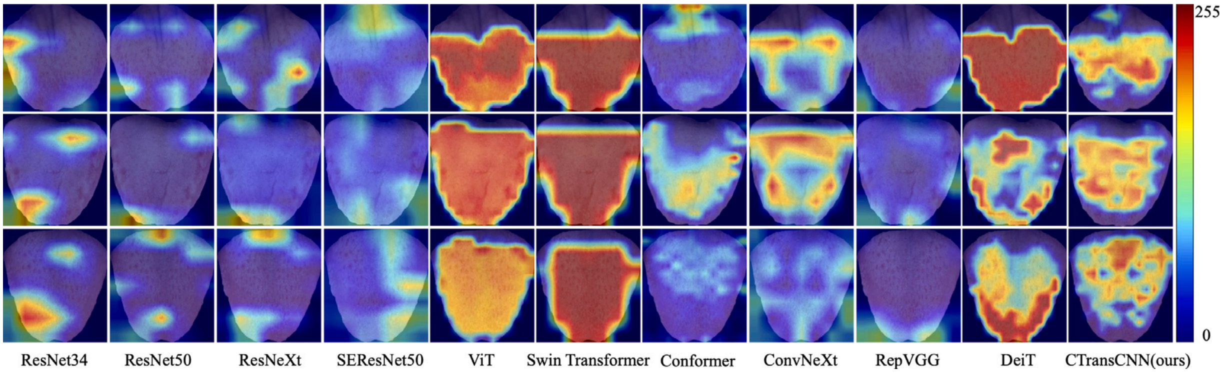


Fig. 11. Visualization results of some samples of the TCMTD by the GradCAM++ method. Warmer heatmap colors (from light blue to dark red) indicate an increase in value or intensity or intensification of a feature.

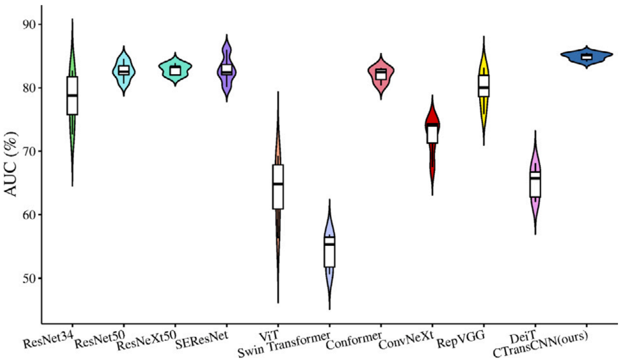


Fig. 12. Violin plots of AUC scores of CTransCNN and 10 comparison networks. A violin plot consists of a central box plot and distribution curves on both sides. The solid black line in the box plot represents the median.

that the model is unstable. The heatmap generated by the CTransCNN model on different images is very similar, indicating that our model is relatively stable. At the same time, our model focuses on both local features and has the global cognitive ability, which better locates the area of interest for classification. Fig. 12 shows the violin plots of AUC scores for the CTransCNN and 10 other comparative networks. ResNet50, SEResNet50, ConvNeXt, and CTransCNN have more concentrated kernel density distributions in their violin plots. The CTransCNN has a higher median, indicating better performance on AUC, while ViT, Swin transformer, and DeiT have poorer AUC performance. We conducted multilabel classification on images using the CTransCNN and ten common networks.

4.7. Ablation study

In order to assess the effectiveness and individual contributions of the various modules in our proposed CTransCNN network, we performed a series of step-by-step ablation experiments on the ChestX-ray11 and TCMTD. We compared the following models to evaluate their performance:

Baseline: The transformer branch is the standard transformer encoder and the CNN branch is the residual block of ResNet, and there is no door-to-door mechanism when information is exchanged and exchanged.

Model 1: Based on the Baseline, the standard transformer encoder is changed to the MMAEF module (without the MSS block).

Model 2: Based on the Baseline, change the residual block of ResNet to rep_method, that is, increase the inner and outer nesting.

Model 3: Based on Model 1, replace the residual block with rep_method.

Model 4: Based on the Baseline, add IIM modules.

Model 5: Based on Model 1, add IIM modules.

Model 6: Based on Model 4, replace the residual block to rep_method.

Model 7: Based on Model 5, replace the residual block with rep_method, and in this case, the MMAEF module does not include the MSS block.

Model 8: Based on Model 6, Model 8 adds the MSS block.

CTransCNN (ours): Based on Model 7, the CTransCNN adds the MSS block.

Quantitative analysis: Tables 7 to 9 provide a quantitative analysis of the experimental results for different modules on the ChestX-ray11 and TCMTD datasets, respectively. In Table 7, for the ChestX-ray11 dataset, compared with the Baseline and Models 1 to 8, CTransCNN achieve AUC improvements of 2.57%, 2.44%, 0.51%, 2.03%, 2.15%, 2.29%, 2.45%, 1.79% and 1.18%. In Table 8, for the TCMTD dataset, compared with the Baseline and Models 1 to 8, CTransCNN achieve AUC improvements of 5.40%, 5.27%, 4.66%, 3.41%, 3.70%, 3.23%, 2.68%, 1.90% and 1.72%. In Table 9, compared with the baseline, the mAP, CF1 and OF1 of the TCMTD were improved by 4.20%, 4.03% and 1.59%, respectively.

Model 1 showed some validity compared to the Baseline, indicating the potential of the MMAEF without the MSS. Model 2 compared with Baseline, and Model 7 compared with Model 5, validating the superiority of rep_method in MBR. Model 4 compared with Baseline, and Model 5 compared with Model 1, demonstrating the validity of IIM. Model 8 compared with Model 6, and CTransCNN compared with Model 7, validating the superiority of the MSS block in the MMAEF. The proposed CTransCNN achieves the best classification results through a good combination of several modules. As can be seen in Tables 7 to 9, each module played a role, confirming the effectiveness of these modules.

Qualitative analysis: Figs. 13 and 14 present the pathological labels and corresponding probability scores for the Baseline and CTransCNN models on the ChestX-ray11 and TCMTD, respectively. The example images and their respective labels in the ChestX-ray11 and TCMTD have one or more labels assigned to each image.

For the ChestX-ray11 dataset, comparing the predicted probabilities of the Baseline and CTransCNN models reveals some differences in various categories. For example, referring to Fig. 13, in the second row and first column, the Baseline model exhibits a relatively low prediction probability for the category ETT-Normal (0.2424). Conversely, in the second row and second column, it demonstrates a higher prediction probability for the same category (0.9971). In contrast, the CTransCNN model assigns prediction probabilities of 0.9910 and 1 to these

Table 7
Comparison of AUC values (%) for our different models in our system on the ChestX-ray11 dataset. The best results are shown in bold.

Network	Baseline	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8	CTransCNN (ours)
ETT-Abnormal	84.69	91.44	86.98	89.80	88.47	91.77	77.34	**93.88**	88.91	90.82
ETT-Borderline	89.32	86.06	84.11	89.62	86.88	88.22	84.33	**90.55**	88.91	88.93
ETT-Normal	97.68	97.23	97.65	97.38	97.68	97.21	97.64	97.46	97.53	**97.80**
NGT-Abnormal	78.91	76.40	84.50	78.00	80.66	78.63	82.61	79.15	83.31	**83.42**
NGT-Borderline	78.47	78.91	83.49	80.08	78.18	77.60	**81.57**	77.15	78.06	81.32
NGT-Incompletely Imaged	93.33	92.63	93.58	93.43	92.94	93.19	92.61	93.03	92.70	**94.53**
NGT-Normal	91.87	91.34	91.41	90.65	91.41	91.76	91.14	91.22	91.72	**92.24**
CVC-Abnormal	60.36	63.56	64.58	63.50	61.84	62.31	63.72	61.33	**65.20**	64.59
CVC-Borderline	58.10	58.80	**62.17**	58.65	58.13	58.38	59.96	59.74	60.11	61.90
CVC-Normal	58.29	58.50	**64.09**	58.01	59.33	57.14	61.11	58.43	59.67	63.32
Swan Ganz Catheter Present	97.75	95.31	**98.87**	95.62	97.92	95.71	98.13	95.44	98.01	98.19
Mean	80.80	80.93	82.86	81.34	81.22	81.08	80.92	81.58	82.19	**83.37**

Table 8
Comparison of AUC values (%) for our different models in our system on the TCMTD dataset. The best results are shown in bold.

Network	Qixu	Qiyu	Shire	Tanshi	Tebing	Xueyu	Yinxu	Mean
Baseline	80.33	**88.31**	82.29	77.27	80.53	83.67	75.72	81.16
Model 1	75.86	78.82	81.47	79.12	94.55	81.61	77.55	81.29
Model 2	81.97	81.18	85.24	82.31	79.70	85.36	77.49	81.90
Model 3	81.08	77.82	84.14	82.07	89.27	**87.86**	79.76	83.15
Model 4	81.14	81.24	78.70	78.69	**96.20**	85.78	78.21	82.86
Model 5	80.08	84.79	82.32	76.37	96.04	84.68	79.02	83.33
Model 6	79.28	84.11	**88.72**	83.67	89.11	84.39	77.87	83.88
Model 7	**83.94**	86.06	85.02	84.10	86.96	86.16	80.33	84.66
Model 8	80.39	82.44	86.82	85.00	91.58	85.08	82.54	84.84
CTransCNN (ours)	83.93	87.55	87.96	**86.53**	91.91	85.36	**82.67**	**86.56**

Table 9
Classification performance (mean±standard deviation) of our different models in our system on the TCMTD. The best results are shown in bold.

Network	mAP (%)	CP (%)	CR (%)	CF1 (%)	OP (%)	OR (%)	OF1 (%)
Baseline	63.47 ± 0.76	58.59 ± 2.31	62.15 ± 0.87	60.29 ± 1.27	78.15 ± 1.22	87.43 ± 1.12	82.53 ± 1.15
Model 1	63.92 ± 1.53	56.62 ± 2.59	64.16 ± 2.21	60.07 ± 0.87	76.20 ± 1.27	89.05 ± 1.16	82.11 ± 0.51
Model 2	63.73 ± 3.10	57.49 ± 4.69	63.68 ± 1.59	60.35 ± 3.00	77.16 ± 2.75	88.87 ± 1.39	82.57 ± 1.74
Model 3	64.92 ± 1.39	57.66 ± 0.79	64.02 ± 0.99	60.66 ± 0.40	76.94 ± 0.46	88.94 ± 1.28	82.50 ± 0.65
Model 4	65.77 ± 1.54	60.49 ± 2.89	63.90 ± 1.01	62.10 ± 1.43	78.30 ± 1.08	89.17 ± 0.59	83.38 ± 0.70
Model 5	65.09 ± 1.56	59.62 ± 2.67	63.37 ± 3.07	61.33 ± 1.34	78.10 ± 1.41	88.38 ± 1.80	82.90 ± 0.75
Model 6	65.86 ± 1.73	58.64 ± 1.77	65.52 ± 1.14	61.88 ± 1.39	77.82 ± 0.88	**90.49 ± 0.95**	83.68 ± 0.88
Model 7	65.22 ± 2.47	58.05 ± 1.28	64.68 ± 1.25	61.17 ± 0.41	76.13 ± 1.23	89.87 ± 0.78	82.42 ± 0.73
Model 8	67.27 ± 1.35	60.87 ± 1.27	65.35 ± 1.34	63.02 ± 0.89	77.97 ± 0.49	90.38 ± 0.52	83.71 ± 0.50
CTransCNN(ours)	**67.67 ± 1.48**	**63.31 ± 4.55**	**65.42 ± 2.44**	**64.32 ± 3.46**	**79.51 ± 1.05**	89.31 ± 0.44	**84.12 ± 0.67**

respective samples. This shows that the CTransCNN model can alleviate the label imbalance problem to a certain extent. The CTransCNN model exhibits higher prediction probabilities compared to the Baseline model, particularly in the ETT-Normal and NGT-Incompletely Imaged categories. This suggests that the CTransCNN model performs better in general and is more accurate in these categories.

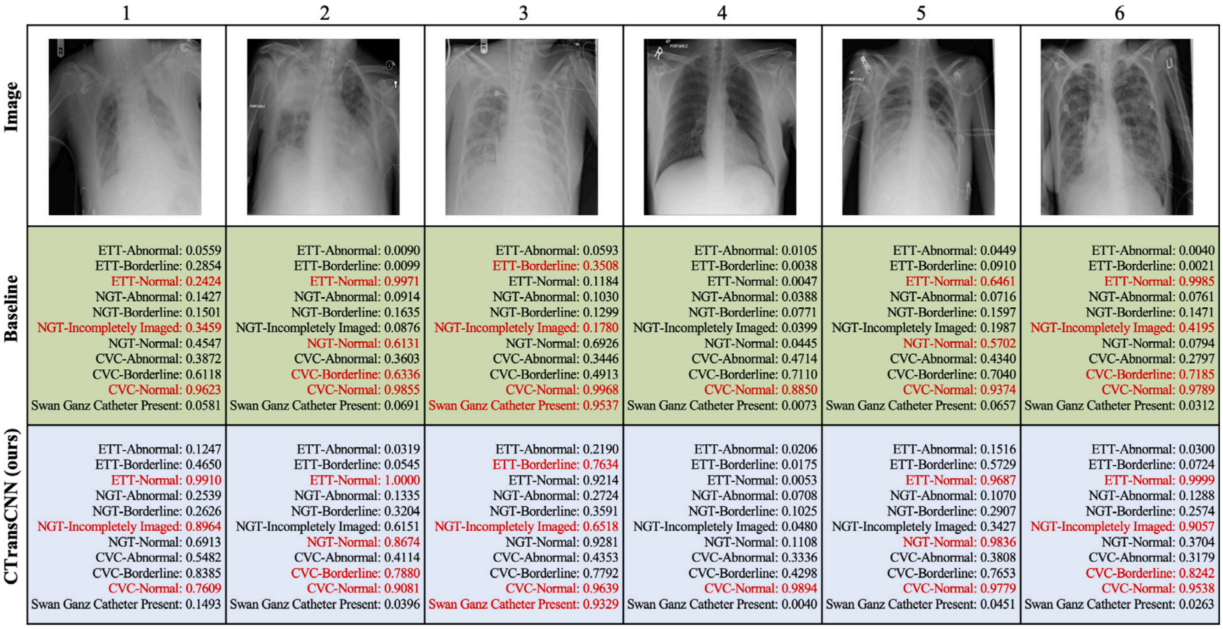
In the TCMTD, the CTransCNN outperforms the Baseline in recognizing the 'Qixu' and 'Tanshi' constitutions while showing similar or slightly improved performance in the 'Shire' and 'Xueyu' constitutions. In the Baseline, for the 'Qixu' constitution, the predicted probability score for this label is close to 1, as evident in the third and eighth columns of Fig. 14. This suggests that the Baseline model tends to prioritize the 'Qixu' constitution over other categories. On the other hand, the CTransCNN does not exhibit this bias. In the case of CTransCNN, specific constitution types yield highly certain prediction results. Here, probability scores for corresponding labels are nearly 1, while scores for other labels approach 0. For example, in the CTransCNN, the probability scores for the Shire and 'Tanshi' constitutions are close to 1, while the scores for other labels are close to 0.

5. Conclusion

The hybrid CNN and transformer architecture for multilabel image classification (CTransCNN) has demonstrated remarkable performance on public datasets (such as ChestX-ray11 and NIH ChestX-ray14), as well as on a private TCMTD multilabel classification dataset. Nonetheless, there are two noteworthy limitations: (i) As indicated by the ablation experiments carried out on the ChestX-ray11 and TCMTD datasets in Section 4.7, the capability of CTransCNN to handle label dependencies might require further improvement. This suggests that when dealing with numerous multilabel categories, fine-tuning the model may be necessary to achieve high accuracy. (ii) The operational speed of the CTransCNN model could potentially face compromises when deployed on specific portable devices.

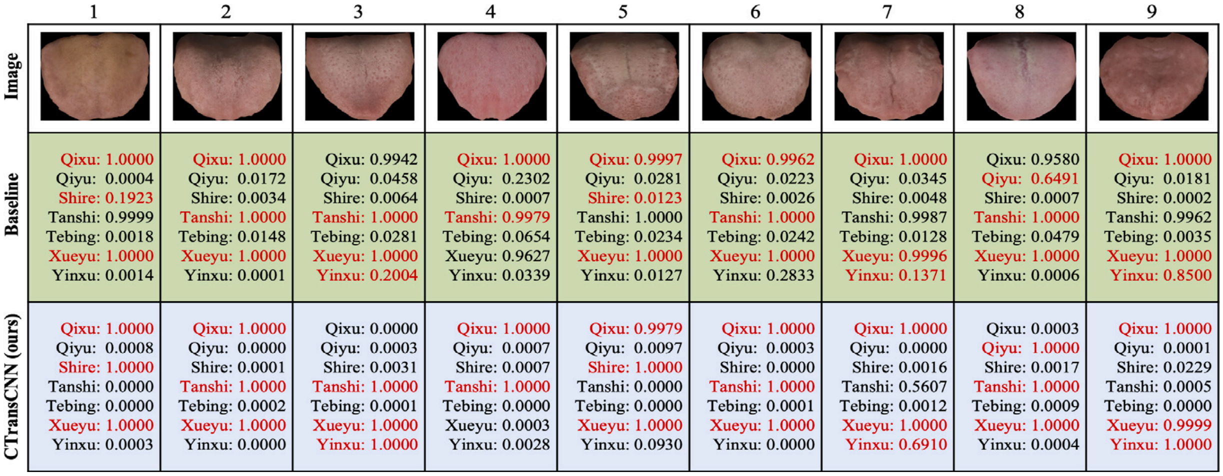
In future research, to enhance data diversity, we will focus on generating high-resolution medical images. Additionally, due to the adoption of a hybrid CNN and transformer structure, there has been a certain increase in the model's computational complexity. Therefore, in our upcoming studies, we will prioritize investigating methods to maintain the multilabel image classification performance while making the proposed model more lightweight. This would render it suitable for deployment on mobile devices, assisting doctors in diagnosis and driving advancements in the field of medicine. Simultaneously, the application of computer-aided diagnosis systems in clinics reduces the workload of doctors to a certain extent and improves diagnostic efficiency [37–39].

In this paper, we introduce the novel CTransCNN model, which integrates image representation features with correlations between medical image labels. By fusing local features and global representations, we capture and explore correlations between labels. We utilize the label



	1	2	3	4	5	6
Baseline	ETT-Abnormal: 0.0559 ETT-Borderline: 0.2854 **ETT-Normal: 0.2424** NGT-Abnormal: 0.1427 NGT-Borderline: 0.1501 **NGT-Incompletely Imaged: 0.3459** NGT-Normal: 0.4547 CVC-Abnormal: 0.3872 CVC-Borderline: 0.6118 **CVC-Normal: 0.9623** Swan Ganz Catheter Present: 0.0581	ETT-Abnormal: 0.0090 ETT-Borderline: 0.0099 **ETT-Normal: 0.9971** NGT-Abnormal: 0.0914 NGT-Borderline: 0.1635 NGT-Incompletely Imaged: 0.0876 **NGT-Normal: 0.6131** CVC-Abnormal: 0.3603 **CVC-Borderline: 0.6336** **CVC-Normal: 0.9855** Swan Ganz Catheter Present: 0.0691	ETT-Abnormal: 0.0593 **ETT-Borderline: 0.3508** ETT-Normal: 0.1184 NGT-Abnormal: 0.1030 NGT-Borderline: 0.1299 **NGT-Incompletely Imaged: 0.1780** NGT-Normal: 0.6926 CVC-Abnormal: 0.3446 CVC-Borderline: 0.4913 **CVC-Normal: 0.9968** **Swan Ganz Catheter Present: 0.9537**	ETT-Abnormal: 0.0105 ETT-Borderline: 0.0038 ETT-Normal: 0.0047 NGT-Abnormal: 0.0388 NGT-Borderline: 0.0771 NGT-Incompletely Imaged: 0.0399 NGT-Normal: 0.0445 CVC-Abnormal: 0.4714 CVC-Borderline: 0.7110 **CVC-Normal: 0.8850** Swan Ganz Catheter Present: 0.0073	ETT-Abnormal: 0.0449 ETT-Borderline: 0.0910 **ETT-Normal: 0.6461** NGT-Abnormal: 0.0716 NGT-Borderline: 0.1597 NGT-Incompletely Imaged: 0.1987 **NGT-Normal: 0.5702** CVC-Abnormal: 0.4340 CVC-Borderline: 0.7040 **CVC-Normal: 0.9374** Swan Ganz Catheter Present: 0.0657	ETT-Abnormal: 0.0040 ETT-Borderline: 0.0021 **ETT-Normal: 0.9985** NGT-Abnormal: 0.0761 NGT-Borderline: 0.1471 **NGT-Incompletely Imaged: 0.4195** NGT-Normal: 0.0794 CVC-Abnormal: 0.2797 **CVC-Borderline: 0.7185** **CVC-Normal: 0.9789** Swan Ganz Catheter Present: 0.0312
CTransCNN (ours)	ETT-Abnormal: 0.1247 ETT-Borderline: 0.4650 **ETT-Normal: 0.9910** NGT-Abnormal: 0.2539 NGT-Borderline: 0.2626 **NGT-Incompletely Imaged: 0.8964** NGT-Normal: 0.6913 CVC-Abnormal: 0.5482 CVC-Borderline: 0.8385 **CVC-Normal: 0.7609** Swan Ganz Catheter Present: 0.1493	ETT-Abnormal: 0.0319 ETT-Borderline: 0.0545 **ETT-Normal: 1.0000** NGT-Abnormal: 0.1335 NGT-Borderline: 0.3204 NGT-Incompletely Imaged: 0.6151 **NGT-Normal: 0.8674** CVC-Abnormal: 0.4114 **CVC-Borderline: 0.7880** **CVC-Normal: 0.9081** Swan Ganz Catheter Present: 0.0396	ETT-Abnormal: 0.2190 **ETT-Borderline: 0.7634** ETT-Normal: 0.9214 NGT-Abnormal: 0.2724 NGT-Borderline: 0.3591 **NGT-Incompletely Imaged: 0.6518** NGT-Normal: 0.9281 CVC-Abnormal: 0.4353 CVC-Borderline: 0.7792 **CVC-Normal: 0.9639** **Swan Ganz Catheter Present: 0.9329**	ETT-Abnormal: 0.0206 ETT-Borderline: 0.0175 ETT-Normal: 0.0053 NGT-Abnormal: 0.0708 NGT-Borderline: 0.1025 NGT-Incompletely Imaged: 0.0480 NGT-Normal: 0.1108 CVC-Abnormal: 0.3336 CVC-Borderline: 0.4298 **CVC-Normal: 0.9894** Swan Ganz Catheter Present: 0.0040	ETT-Abnormal: 0.1516 ETT-Borderline: 0.5729 **ETT-Normal: 0.9687** NGT-Abnormal: 0.1070 NGT-Borderline: 0.2907 NGT-Incompletely Imaged: 0.3427 **NGT-Normal: 0.9836** CVC-Abnormal: 0.3808 CVC-Borderline: 0.7653 **CVC-Normal: 0.9779** Swan Ganz Catheter Present: 0.0451	ETT-Abnormal: 0.0300 ETT-Borderline: 0.0724 **ETT-Normal: 0.9999** NGT-Abnormal: 0.1288 NGT-Borderline: 0.2574 **NGT-Incompletely Imaged: 0.9057** NGT-Normal: 0.3704 CVC-Abnormal: 0.3179 **CVC-Borderline: 0.8242** **CVC-Normal: 0.9538** Swan Ganz Catheter Present: 0.0263

Fig. 13. Examples of the recognition results on the ChestX-ray11 dataset. This presents 11 predicted pathological labels along with their corresponding probability scores. The ground true labels are highlighted in red for emphasis.



	1	2	3	4	5	6	7	8	9
Baseline	**Qixu: 1.0000** Qiyu: 0.0004 **Shire: 0.1923** Tanshi: 0.9999 Tebing: 0.0018 **Xueyu: 1.0000** Yinxu: 0.0014	**Qixu: 1.0000** Qiyu: 0.0172 Shire: 0.0034 **Tanshi: 1.0000** Tebing: 0.0148 **Xueyu: 1.0000** Yinxu: 0.0001	Qixu: 0.9942 Qiyu: 0.0458 Shire: 0.0064 **Tanshi: 1.0000** Tebing: 0.0281 **Xueyu: 1.0000** **Yinxu: 0.2004**	**Qixu: 1.0000** Qiyu: 0.2302 Shire: 0.0007 **Tanshi: 0.9979** Tebing: 0.0654 Xueyu: 0.9627 Yinxu: 0.0339	**Qixu: 0.9997** Qiyu: 0.0281 **Shire: 0.0123** Tanshi: 1.0000 Tebing: 0.0234 **Xueyu: 1.0000** Yinxu: 0.0127	**Qixu: 0.9962** Qiyu: 0.0223 Shire: 0.0026 **Tanshi: 1.0000** Tebing: 0.0242 **Xueyu: 1.0000** Yinxu: 0.2833	**Qixu: 1.0000** Qiyu: 0.0345 Shire: 0.0048 Tanshi: 0.9987 Tebing: 0.0128 **Xueyu: 0.9996** **Yinxu: 0.1371**	Qixu: 0.9580 **Qiyu: 0.6491** Shire: 0.0007 **Tanshi: 1.0000** Tebing: 0.0479 **Xueyu: 1.0000** Yinxu: 0.0006	**Qixu: 1.0000** Qiyu: 0.0181 Shire: 0.0002 Tanshi: 0.9962 Tebing: 0.0035 **Xueyu: 1.0000** **Yinxu: 0.8500**
CTransCNN (ours)	**Qixu: 1.0000** Qiyu: 0.0008 **Shire: 1.0000** Tanshi: 0.0000 Tebing: 0.0000 **Xueyu: 1.0000** Yinxu: 0.0003	**Qixu: 1.0000** Qiyu: 0.0000 Shire: 0.0001 **Tanshi: 1.0000** Tebing: 0.0002 **Xueyu: 1.0000** Yinxu: 0.0000	Qixu: 0.0000 Qiyu: 0.0003 Shire: 0.0031 **Tanshi: 1.0000** Tebing: 0.0001 **Xueyu: 1.0000** **Yinxu: 1.0000**	**Qixu: 1.0000** Qiyu: 0.0007 Shire: 0.0007 **Tanshi: 1.0000** Tebing: 0.0000 Xueyu: 0.0003 Yinxu: 0.0028	**Qixu: 0.9979** Qiyu: 0.0097 **Shire: 1.0000** Tanshi: 0.0000 Tebing: 0.0000 **Xueyu: 1.0000** Yinxu: 0.0930	**Qixu: 1.0000** Qiyu: 0.0003 Shire: 0.0000 **Tanshi: 1.0000** Tebing: 0.0001 **Xueyu: 1.0000** Yinxu: 0.0000	**Qixu: 1.0000** Qiyu: 0.0000 Shire: 0.0016 Tanshi: 0.5607 Tebing: 0.0012 **Xueyu: 1.0000** **Yinxu: 0.6910**	Qixu: 0.0003 **Qiyu: 1.0000** Shire: 0.0017 **Tanshi: 1.0000** Tebing: 0.0009 **Xueyu: 1.0000** Yinxu: 0.0004	**Qixu: 1.0000** Qiyu: 0.0001 Shire: 0.0229 Tanshi: 0.0005 Tebing: 0.0000 **Xueyu: 0.9999** **Yinxu: 1.0000**

Fig. 14. Example of recognition results for the TCMTD. This shows 7 predicted pathological labels and their respective probability scores. True labels are highlighted in red to draw attention. Due to the limitation of sample collection, the ‘Pinghe’ and ‘Yangxu’ conditions of the nine TCM constitutions are not shown.

embedding and the MSS block of the MMAEF to investigate the hidden connections among disease complications and label differences, while the MBR module is employed to optimize the model and effectively reduce the number of parameters. Furthermore, the information interaction modules, specifically the C2T module and T2C module, facilitate feature transfer between the two branches and introduce nonlinearity. These modules play a crucial role in achieving successful multilabel image classification tasks. Experimental results demonstrate that the proposed CTransCNN exhibits heightened efficacy in discerning multilabel images and extracting more intricate information. It achieves superior results across the metrics evaluated on the three investigated datasets. Collectively, these findings underscore the network's exceptional performance in the realm of medical multilabel image classification, with strong generalization ability that can be applied to other medical multilabel image classification tasks.

We confirm that the manuscript is not currently under consideration or published in another journal. All authors declare that they have no known competing financial interests or personal relationships that could affect the work reported in this article. The authors declare no potential conflicts of interest concerning the research, authorship or publication of this article.

CRediT authorship contribution statement

Xin Wu: Conceptualization, Writing – original draft & review & editing, Data collection, Methodology, Formal analysis, Visualization, Validation. **Yue Feng:** Conceptualization, Investigation, Writing – original draft & review & editing, Supervision, Project administration, Funding acquisition. **Hong Xu:** Writing – review & editing, Funding acquisition. **Zhuosheng Lin:** Writing – review & editing. **Tao Chen:** Writing – review & editing. **Shengke Li:** Writing – review & editing, Validation. **Shihan Qiu:** Writing – review & editing. **Qichao Liu:** Validation. **Yuangang Ma:** Validation. **Shuangsheng Zhang:** Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

We are open to sharing our implementation code and additional details with researchers interested in our study. For access to the code or further information, please contact the corresponding author.

Acknowledgments

This work is supported by the Basic Research and Applied Basic Research Key Project in General Colleges and Universities of Guangdong Province, China (2021ZDZX1032); the Special Project of Guangdong Province, China (2020A13 13030021); and the Scientific Research Project of Wuyi University (2018TP023, 2018GR003). (Xin Wu and Yue Feng contributed equally to this article.) We extend our gratitude to all colleagues who provided valuable suggestions and support throughout this research. We are open to sharing our implementation code and additional details with researchers interested in our study. For access to the code or further information, please contact the corresponding author.

References

[1] Y.-W. Lee, S.-K. Huang, R.-F. Chang, CheXGAT: A disease correlation-aware network for thorax disease diagnosis from chest X-ray images, Artif. Intell. Med. 132 (2022) 102382, http://dx.doi.org/10.1016/j.artmed.2022.102382.

[2] A. Majkowska, S. Mittal, D.F. Steiner, J.J. Reicher, S.M. McKinney, G.E. Duggan, K. Eswaran, P.-H.C. Chen, Y. Liu, S.R. Kalidindi, A. Ding, G.S. Corrado, D. Tse, S. Shetty, Chest radiograph interpretation with deep learning models: Assessment with radiologist-adjudicated reference standards and population-adjusted evaluation, Radiology 294 (2) (2020) 421–431, http://dx.doi.org/10.1148/radiol.2019191293.

[3] Y. Gong, Y. Jia, T. Leung, A. Toshev, S. Ioffe, Deep convolutional ranking for multilabel image annotation, 2013, http://dx.doi.org/10.48550/arXiv.1312.4894, arXiv preprint arXiv:1312.4894.

[4] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, HCP: A flexible CNN framework for multi-label image classification, IEEE Trans. Pattern Anal. Mach. Intell. 38 (9) (2016) 1901–1907, http://dx.doi.org/10.1109/TPAMI.2015.2491929.

[5] J. Wang, L. Yang, Z. Huo, W. He, J. Luo, Multi-label classification of fundus images with EfficientNet, IEEE Access 8 (2020) 212499–212508, http://dx.doi.org/10.1109/ACCESS.2020.3040275.

[6] W. Shi, X. Liu, Q. Yu, Correlation-aware multi-label active learning for web service tag recommendation, in: 2017 IEEE International Conference on Web Services (ICWS), Honolulu, HI, 2017, pp. 229–236, http://dx.doi.org/10.1109/ICWS.2017.37.

[7] X. Cheng, H. Lin, X. Wu, D. Shen, F. Yang, H. Liu, N. Shi, MLTR: Multi-label classification with transformer, in: 2022 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6, http://dx.doi.org/10.1109/ICME52920.2022.9860016.

[8] Z. Wang, T. Chen, G. Li, R. Xu, L. Lin, Multi-label image recognition by recurrently Discovering Attentional Regions, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, http://dx.doi.org/10.1109/iccv.2017.58.

[9] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, http://dx.doi.org/10.48550/arXiv.1409.1556, arXiv preprint arXiv:1409.1556.

[10] L. Nie, T. Chen, Z. Wang, W. Kang, L. Lin, Multi-label image recognition with attentive transformer-localizer module, Multimedia Tools Appl. 81 (6) (2022) 7917–7940, http://dx.doi.org/10.1007/s11042-021-11818-8.

[11] Z.M. Chen, Q. Cui, B. Zhao, R. Song, X. Zhang, O. Yoshie, SST: Spatial and semantic transformers for multi-label image recognition, IEEE Trans. Image Process. 31 (2022) 2570–2583, http://dx.doi.org/10.1109/TIP.2022.3148867.

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, An image is worth 16x16 words: Transformers for image recognition at scale, 2020, http://dx.doi.org/10.48550/arXiv.2010.11929, arXiv preprint arXiv:2010.11929.

[13] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, 2016, pp. 770–778, http://dx.doi.org/10.1109/cvpr.2016.90.

[14] L. Song, J. Liu, B. Qian, M. Sun, K. Yang, M. Sun, S. Abbas, A deep multi-modal CNN for multi-instance multi-label image classification, IEEE Trans. Image Process. 27 (12) (2018) 6025–6038, http://dx.doi.org/10.1109/TIP.2018.2864920.

[15] I. Allaouzi, M.B. Ahmed, A novel approach for multi-label chest X-Ray classification of common Thorax diseases, IEEE Access 7 (2019) 64279–64288, http://dx.doi.org/10.1109/ACCESS.2019.2916849.

[16] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, W. Xu, Cnn-rnn: A unified framework for multi-label image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016, pp. 2285–2294, http://dx.doi.org/10.1109/cvpr.2016.251.

[17] S. Taslimi, S. Taslimi, N. Fathi, M. Salehi, M.H. Rohban, SwinCheX: Multi-label classification on chest X-ray images with transformers, 2022, http://dx.doi.org/10.48550/arXiv.2206.04246, arXiv preprint arXiv:2206.04246.

[18] J. Lanchantin, T. Wang, V. Ordonez, Y. Qi, General multi-label image classification with transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16478–16488, http://dx.doi.org/10.1109/cvpr46437.2021.01621, Online.

[19] X. Zhu, J. Cao, J. Ge, W. Liu, B. Liu, Two-stream transformer for multi-label image classification, in: Proceedings of the 30th ACM International Conference on Multimedia, Association for Computing Machinery, New York, NY, USA, 2022, pp. 3598–3607, http://dx.doi.org/10.1145/3503161.3548343, Online (2022).

[20] T. Ridnik, E. Ben-Baruch, N. Zamir, A. Noy, I. Friedman, M. Protter, L. Zelnik-Manor, Asymmetric loss for multi-label classification, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, Canada, 2021, pp. 82–91, http://dx.doi.org/10.1109/iccv48922.2021.00015.

[21] L. Yi, L. Zhang, X. Xu, J. Guo, Multi-label softmax networks for pulmonary nodule classification using unbalanced and dependent categories, IEEE Trans. Med. Imaging 42 (1) (2023) 317–328, http://dx.doi.org/10.1109/TMI.2022.3211085.

[22] Z. Yan, W. Liu, S. Wen, Y. Yang, Multi-label image classification by feature attention network, IEEE Access 7 (2019) 98005–98013, http://dx.doi.org/10.1109/ACCESS.2019.2929512.

[23] W. Zhou, Y. Hou, D. Chen, H. Hu, T. Su, Attention-augmented memory network for image multi-label classification, ACM Trans. Multimedia Comput. Commun. Appl. 19 (3) (2023) 116, http://dx.doi.org/10.1145/3570166.

[24] S. Liu, L. Zhang, X. Yang, H. Su, J. Zhu, Query2label: A simple transformer way to multi-label classification, 2021, http://dx.doi.org/10.48550/arXiv.2107.10834, arXiv preprint arXiv:2107.10834.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30 (NIPS 2017), in: Advances in Neural Information Processing Systems, vol. 30, Long Beach, CA, 2017, http://dx.doi.org/10.48550/arXiv.1706.03762.

[26] G. Patterson, J. Hays, COCO attributes: Attributes for people, animals, and objects, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 85–100, http://dx.doi.org/10.1007/978-3-319-46466-4_6.

[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, Focal loss for dense object detection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice,Italy, 2017, http://dx.doi.org/10.1109/iccv.2017.324.

[28] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, R.M. Summers, Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, 2017, pp. 2097–2106, http://dx.doi.org/10.1109/cvpr.2017.369.

[29] H. Liu, Y. Feng, H. Xu, S. Liang, H. Liang, S. Li, J. Zhu, S. Yang, F. Li, MEA-Net: multilayer edge attention network for medical image segmentation, Sci. Rep. 12 (1) (2022) 7868, http://dx.doi.org/10.1038/s41598-022-11852-y.

[30] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, 2017, http://dx.doi.org/10.48550/arXiv.1711.05101, arXiv preprint arXiv:1711.05101.

[31] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, J. Sun, Repvgg: Making vgg-style convnets great again, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13733–13742, http://dx.doi.org/10.1109/cvpr46437.2021.01352, Online (2021).

[32] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, 2017, pp. 1492–1500, http://dx.doi.org/10.1109/cvpr.2017.634.

[33] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, USA, 2018, pp. 7132–7141, http://dx.doi.org/10.1109/cvpr.2018.00745.

[34] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, H. Jegou, Training data-efficient image transformers &; distillation through attention, in: M. Meila, T. Zhang (Eds.), Proceedings of the 38th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 139, PMLR, 2021, pp. 10347–10357, URL https://proceedings.mlr.press/v139/touvron21a.html.

[35] Z. Peng, Z. Guo, W. Huang, Y. Wang, L. Xie, J. Jiao, Q. Tian, Q. Ye, Conformer: Local features coupling global representations for recognition and detection, IEEE Trans. Pattern Anal. Mach. Intell. (2023) 1–15, http://dx.doi.org/10.1109/TPAMI.2023.3243048.

[36] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, S. Xie, A convnet for the 2020s, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, 2022, pp. 11976–11986, http://dx.doi.org/10.48550/arxiv.2201.03545.

[37] 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 Syst. Appl. 229 (2023) 120450, http://dx.doi.org/10.1016/j.eswa.2023.120450.

[38] Y. Luo, Q. Huang, L. Liu, Classification of tumor in one single ultrasound image via a novel multi-view learning strategy, Pattern Recognit. (2023) 109776, http://dx.doi.org/10.1016/j.patcog.2023.109776.

[39] Y. Luo, Z. Lu, L. Liu, Q. Huang, Deep fusion of human-machine knowledge with attention mechanism for breast cancer diagnosis, Biomed. Signal Process. Control 84 (2023) 104784, http://dx.doi.org/10.1016/j.bspc.2023.104784.