

# Efficient Channel Transformer for Processing Agricultural Aerial Images

Guangquan Lu, Shilong Lin, Shichao Zhang, Jiacheng Jiang\*, Yadan Han\*,  
Guoqiu Wen, and Taotao Qiu

**Abstract:** Existing methods always neglect the value of channel features in processing multi-modal agricultural aerial images with Near-Infrared (NIR) characteristics, as well as ignore the importance of channel features in fusion features that contain both downsampled low-level features and upsampled high-level features. Two modules are proposed in this paper to address this issue. For fusion features composed of spatial information, semantic information, and multi-modal information, a transformer-based channel feature enhancement module is first constructed to facilitate the recognition of fusion features located in different channels. The second module is Dual Cross-Entropy Dice (Dual-CE-Dice) loss, which can improve the phenomenon of class imbalance while helping the model to better learn channel features. Extensive experiments have been conducted on the Agriculture-Vision-2021 dataset and the Tianchi suichang-round1 dataset, proving that the proposed method Channel Transformer (CFormer) is superior to the previous methods.

**Key words:** image segmentation; agriculture; multi-modal images

## 1 Introduction

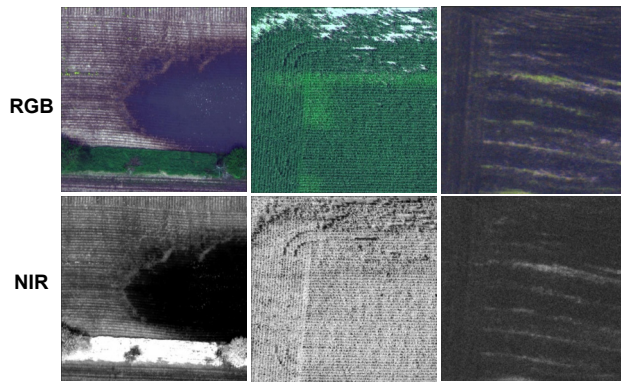
Remote sensing image segmentation is a branch of semantic segmentation tasks, which is of great significance for studying the state of the geographical environment and traces of human activities. Its purpose is to classify each pixel in a remote sensing image into a semantic class. In recent years, this task has been used in many real-world scenarios, such as urban

planning<sup>[1]</sup>, disaster assessment<sup>[2]</sup>, land cover<sup>[3]</sup>, etc. Agriculture is a field closely related to human beings. Remote sensing segmentation tasks can help land use and cover classification, crop monitoring and evaluation, and pest detection. This can bring huge benefits to human society, but there is still little research on the remote sensing segmentation task of purely agricultural aerial images. Agricultural aerial images have certain particularities, which contain Near-Infrared (NIR) features captured by specific cameras, as shown in Fig. 1, making agricultural aerial images have multi-modal features. Chiu et al.<sup>[4]</sup> proved that NIR features can be concatenated with Red-Green-Blue (RGB) features to form RGB-Near-infrared (RGBN) images for training semantic segmentation models. However, in the field of remote sensing image semantic segmentation, most models are designed based on RGB images. The mechanical way of using them directly to train RGBN images is not enough to make full use of the multi-modal features stitched along the channel, and channel enhancement method can be one of the key techniques.

- Guangquan Lu, Shilong Lin, Shichao Zhang, Jiacheng Jiang, Yadan Han, and Guoqiu Wen are with the Key Lab of Education Blockchain and Intelligent Technology, Ministry of Education, Guangxi Normal University, Guilin 541004, China, and also with Guangxi Key Lab of Multi-Source Information Mining and Security, Guangxi Normal University, Guilin 541004, China. E-mail: lugq@mailbox.gxnu.edu.cn; shilonglin@stu.gxnu.edu.cn; zhangsc@csu.edu.cn; jicj9927@163.com; 982724943@qq.com; wenguoqiu2008@163.com.
- Taotao Qiu is with College of Physical Education and Health, Guangxi Normal University, Guilin 541004, China. E-mail: gxqiuataotao@mailbox.gxnu.edu.cn.

\* To whom correspondence should be addressed.

Manuscript received: 2024-08-29; revised: 2025-01-10;  
accepted: 2025-03-27



**Fig. 1** RGB and NIR bands in the agricultural remote sensing image dataset Agriculture-Vision-2021 serve as an example, where RGB and NIR impart multi-modal properties to the image.

In recent years, many researchers have made outstanding contributions to channel attention, and these methods have been used in the semantic segmentation of remote sensing images. SENet<sup>[5]</sup> first proposed the concept of channel attention, which proposed to use a global average pooling and a structure similar to a multi-layer perceptron to extract features on the channel. The subsequently proposed CBAM<sup>[6]</sup> pair considers that the global average pooling in SENet will limit the channel features, so a maximum pooling layer is introduced to represent the channel features further. Similar studies include BAM<sup>[7]</sup>, ECA-Net<sup>[8]</sup>, etc. These methods are often used on the encoder. Sheng et al.<sup>[9]</sup> used the channel attention method based on the same principle as SENet to enhance the channel information in the downsampled features. SGFNet<sup>[10]</sup> also introduces the same channel attention method as SENet in the process of downsampling to enhance the channel representation ability of the encoder. Panboonyuen et al.<sup>[11]</sup> used channel attention similar to the Multilayer Perceptron (MLP) structure of SENet in the downsampling process of the remote sensing image segmentation model. A similar approach is LANet<sup>[12]</sup>, which proposed a method for channel enhancement of features in batches. The channel attention it designs also uses the MLP structure to extract channel features of remote sensing images. However, channel attention can be used not only in the down-sampling step of the encoder but also in the upsampling step of the decoder.

However, there are relatively few studies on applying channel attention to the decoder, and even most of the methods directly use existing methods on the decoder. FaPN<sup>[13]</sup> proposes a Feature Selection Module (FSM)

for channel enhancement of downsampled low-level features, which takes inspiration from SE<sup>[5]</sup>. DPFANet<sup>[14]</sup> concatenates adjacent upsampled features on channels in the decoder, and then uses the channel attention method with the same principle as SE to screen important channel features. LASNet<sup>[15]</sup> utilizes the attention method CBAM to enhance the down-sampling features before feature fusion at the decoder, which improves the channel representation ability of the decoder. ERDUnet<sup>[16]</sup> also utilizes CBAM to enhance channel and spatial information in up-sampled features. These methods directly transfer the existing attention method without improving the attention method itself, and they all ignore the importance of fusion features in the decoder. The fusion feature is composed of up-sampled high-level features and down-sampled low-level features. Massive information will lead to the introduction of too many redundant features and noise. An efficient channel attention suppresses the interference of redundant features on model training.

There is a similar phenomenon in the semantic segmentation task of remote sensing images. SCAttNet<sup>[17]</sup> designed a method similar to CBAM<sup>[6]</sup> and used it in the decoder to help the network extract more valuable semantic information in high-resolution aerial images. Similar researches<sup>[18–20]</sup> used channel attention only for high-level features or low-level features before feature aggregation. In addition, TFNet<sup>[21]</sup> is different from previous methods, it proposes a transformer-based channel attention mechanism, but it still retains the attention calculation at the feature map level, which will increase the number of model parameters and computational complexity. Agricultural aerial images may contain multi-modal features such as NIR features, so the fusion features will contain spatial information, semantic information, and multi-modal information in low-level features and high-level features. A large amount of information is gathered in the fusion features, which inevitably leads to the generation of redundant features. Therefore, the channel enhancement method should be introduced to further extract the fusion features, emphasizing the important feature layers in the massive features, and suppressing redundant feature maps while accurately assigning them. The previous channel enhancement methods often contain only simple Convolution (Conv) operations and cannot make full use of the feature mapping containing complex information such as

fusion features. Therefore, a more efficient transformer-based approach (Channel Enhancement Block, CEB) is proposed to enhance the fusion features so that the segmentation head can classify pixels more accurately.

In addition, there are often more obvious class imbalances in semantic segmentation tasks<sup>[22]</sup> and remote sensing image retrieval datasets<sup>[23]</sup>. It is no exception in the agricultural remote sensing image semantic segmentation dataset. In response to this situation, measures must be taken to optimize the model to achieve better performance.

In response to the above problems, we mainly propose solutions based on three ideas: First, compared with the semantic segmentation model based on RGB image design, channel enhancement technology will become a key method in the scene of images with multi-modal features to make better use of channel features. Secondly, unlike the previous channel enhancement methods based on encoder design, we believe that the fusion features in the decoder also have the application value of such methods. Finally, class imbalance should become a topic that cannot be ignored, and certain solutions need to be adopted to improve it.

In summary, the contributions of this paper are as follows:

- A transformer-based channel feature enhancement module that helps enhance the fusion features in the decoder, which is consist of upsampled high-level features and downsampled low-level features.
- A novel Dual Cross-Entropy Dice (Dual-CE-Dice) loss that helps the model learn channel features better while improving the negative impact caused by class imbalance.

Finally, extensive comparative experiments and ablation studies are conducted on the Agriculture-Vision-2021 dataset<sup>[4]</sup> and the Tianchi suichang-round1 dataset<sup>[24]</sup> to validate the above two contributions. To explore the impact of our methods on extracting multi-modal features, experiments are conducted on both RGB and RGBN images, demonstrating the effectiveness and robustness of the proposed model CFormer.

## 2 Related Work

### 2.1 Semantic segmentation of RSIs

Since FCN<sup>[25]</sup> was pioneered in the semantic

segmentation task, the Convolutional Neural Network (CNN) once became the mainstream of the semantic segmentation model<sup>[26–28]</sup>. Due to the effectiveness of these methods, many methods have emerged that can be applied to Remote Sensing Images (RSIs). DeepLab<sup>[29]</sup> is one of the first semantic segmentation methods that proposed to use atrous convolutions to alleviate the limitations of traditional convolution operations. Since then, on this basis, a series of methods based on the Atrous Spatial Pyramid Pooling (ASPP) module have been proposed<sup>[30, 31]</sup>.

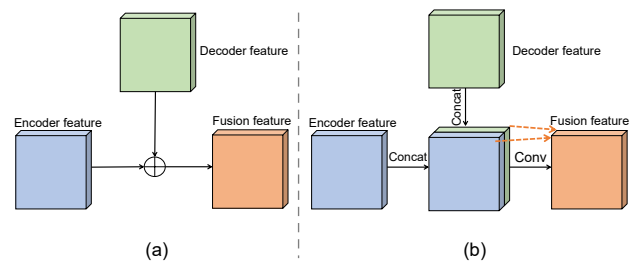
While different models achieve better performance, the computational cost is inevitably rising. Pan et al.<sup>[32]</sup> build modules by integrating Instance Normalization (IN)<sup>[33]</sup> and Batch Normalization (BN)<sup>[34]</sup> in order to improve the performance of deep networks without increasing the computational cost. The integrated module of IN and BN can help the model learn appearance features and content information. This module can be encapsulated into different advanced deep networks, such as their proposed Instance-Batch Normalization Networks (IBN-Nets). On the Agriculture-Vision<sup>[4]</sup> dataset, Yang et al.<sup>[35]</sup> verified that one type of IBN-Net, IBN-Net101-a, could significantly reduce the feature divergence between RGB images and NIR images, and used Switchable Normalization (SN)<sup>[36]</sup> to replace IN and BN in IBN-Net101-a to better help the model utilize them. MSCG-Net<sup>[37]</sup> proposed a multi-scale image training model based on the Self-Constructing Graph (SCG) module, which believes that agricultural remote sensing images of different scales contain different detailed features. Another similar approach is AgriSegNet<sup>[38]</sup>, which proposed a multi-scale image training model based on DeepLabV3+. Many State-Of-The-Art (SOTA) models in the field of semantic segmentation use transformer-based frameworks, such as SegFormer<sup>[39]</sup>, TransUnet<sup>[40]</sup>, etc., they use a transformer as the encoder to extract features, but they often have a disadvantage, that is, a large number of parameters and high computational complexity. It may not be conducive to the landing of semantic segmentation tasks of agricultural remote sensing images. These multi-scale and transformer-based approaches undoubtedly increase training costs, which means larger Graphics Processing Unit (GPU) memory is required. In addition, all of these methods ignore how to fully utilize the importance of channel features in agricultural multi-modal remote sensing image scenes.

## 2.2 Channel attention

In the research direction of channel information enhancement, researchers have proposed a series of channel attention methods. Hu et al.<sup>[5]</sup> first proposed Squeeze-and-Excitation (SE), aiming to use a structure similar to MLP to enhance channel features on features from the residual path. Since the global pooling module in SE may limit the expression ability of channel attention, CBAM<sup>[6]</sup> introduces max pooling on this basis to obtain different representation effects. Similar to the above works are SKNet<sup>[41]</sup> and BAM<sup>[7]</sup>, etc., which are basically the same as SE in the structure of extracting channel attention. Wang et al.<sup>[8]</sup> believed that the two fully connected layers used in SE would introduce a large number of parameters and calculations, so they only used a one-dimensional convolutional layer with a fixed convolution kernel size to replace the fully connected layer structure in SE.

The above methods can be used for both the encoder and decoder, although they are predominantly utilized to enhance channel features within the encoder module. Therefore, in terms of the decoder module, there have been related researches that specifically apply channel enhancement methods. Inspired by SE, FaPN<sup>[13]</sup> leverages the FSM to enhance low-level features obtained through down-sampling, enriching spatial information while suppressing redundant feature maps, subsequently integrating them with high-level features containing abundant semantic information. However, in terms of implementation details, FSM employs solely global average pooling and one-dimensional convolutional layers for channel feature extraction. On the contrary, Peng and Ma<sup>[20]</sup> used channel attention to enhance the high-level features of upsampling, emphasizing the enhancement of semantic information. The method itself is similar to FSM and only involves simple global average pooling and convolution processing. In another of their work<sup>[18]</sup>, the method was once again employed to enhance channel features, using only global average pooling and a single convolutional operation. Khanh et al.<sup>[19]</sup> focused on the low-level features from the encoder and the high-level features from the decoder at the same time, and used the same channel enhancement method to enhance these features. They proposed Channel Attention Gate (CAG), similar to CBAM, simultaneously extracting global average pooling and maximum pooling vectors, and then performing one-dimensional convolution processing for feature extraction.

Apart from the aforementioned works that exclusively employ convolutional layers for channel feature extraction, some works leverage self-attention for channel feature enhancement. DANet<sup>[42]</sup> adopts a self-attention-inspired approach wherein the feature map is initially flattened into vectors, serving as query, key, and value inputs for self-attention computation. This process culminates in an attention map of the same spatial dimensions as the original feature map. CG-Net<sup>[43]</sup>, akin to DANet, incorporates self-attention with the distinction that CG-Net employs multi-head mechanisms to partition feature vectors. After acquiring new features with matching dimensions to the original map, CG-Net employs average pooling to acquire channel features for subsequent enhancement. TFNet<sup>[21]</sup> proposes utilizing attention modules with a transformer structure to extract channel features. Initially, it preserves the multi-head self-attention computation at the feature map level. After modeling contextual relationships, it employs a method similar to CBAM<sup>[6]</sup> to extract channel features. The distinction lies in maintaining a multi-head mechanism during channel feature extraction. The aforementioned works retain attention computation at the feature map level, potentially resulting in increased computational complexity and parameter quantity. Furthermore, they often focus solely on low-level features obtained through down-sampling or high-level features acquired through up-sampling, or a combination of both. However, they overlook enhanced fusion features, making them potentially unsuitable for feature aggregation. Fusion features are often obtained in two ways, as shown in Fig. 2. One way is to add the encoder features and decoder features together. Another way is to concatenate the encoder features and



**Fig. 2 Two common approaches to obtaining examples of fusion features: (a) Adding the encoder features and decoder features; (b) expanding the number of channels by splicing the encoder features and decoder features to twice, then using the Conv layer to compress the number of channels to make it consistent with the original number of feature channels.**

decoder features, expanding the number of channels to twice, and then using convolutional layers to compress the number of channels to match the original feature channels. In the context of agricultural multi-modal remote sensing segmentation tasks, fusion features encompass more spatial and semantic information, along with multi-modal features like NIR features. Effectively enhancing channel features in fusion features can circumvent feature redundancy stemming from excessive features.

### 3 Method

#### 3.1 Overall architecture

The overall structure of the proposed method Channel Transformer (CFormer) is shown in Fig. 3. Yang et al.<sup>[35]</sup> verified that using IBN-Net101-a (IBN-Net101-a) in ResNet-101<sup>[44]</sup> and applying it as an encoder to the DeepLabV3+<sup>[31]</sup> model can effectively reduce the feature divergence between RGB images and NIR images. We also use IBN-Net101-a as the encoder. First, the ASPP module of DeepLabV3+ was replaced with the Pyramid Pooling Module (PPM)<sup>[45]</sup>. Because PPM performs pooling operations on feature maps at different pyramid levels, it captures contextual information at various scales. In contrast, while ASPP's individual atrous convolution kernels have different dilation rates, they might still struggle to capture multi-scale information effectively. Secondly, it is believed that the features in the DeepLabV3+ decoder may not be able to better achieve the purpose of contextual interaction. Inspired by FPN<sup>[46]</sup>, we use feature  $C_i \in \{C_2, C_3, C_4, C_5\}$  from the encoder and feature  $B_i \in \{B_2, B_3, B_4, B_5\}$  from the decoder for

feature fusion, where  $C_i \in \mathbb{R}^{d_i \times \frac{H}{2^i} \times \frac{W}{2^i}}$ ,  $H \times W$  represents the spatial dimension of the input image,  $d_i$  represents the number of channels,  $i$  in  $C_i$  and  $B_i$  represents the  $i$ -th level feature in the encoder or decoder. Similar to the concept of the DeepLabV3+ decoder, all  $B_i$  have the same scale with  $C_2$ , that is,  $B_i \in \mathbb{R}^{d_2 \times \frac{H}{4} \times \frac{W}{4}}$ . The feature fusion formula is defined as Eq. (1),

$$B_i = C_i + B_{i+1} \quad (1)$$

where  $i$  represents a certain level of features from the encoder or decoder. The fusion feature  $B_i \in \{B_2, B_3, B_4, B_5\}$  can be obtained via Eq. (1). We will introduce how to perform feature enhancement on the fusion feature  $B_i$  in subsequent chapters.

#### 3.2 Channel transformer

The architecture of the CEB we design is shown in Fig. 4c. First, the fusion feature  $B_i \in \mathbb{R}^{d_2 \times \frac{H}{4} \times \frac{W}{4}}$  gets the channel vector  $h_i \in \mathbb{R}^{d_2 \times 1 \times 1}$  through the CTM.  $h_i$  contains rich channel features from  $B_i$ , and then uses the form of  $h_i \times B_i$  to perform channel enhancement on  $B_i$ . We denote the product result as an enhanced feature  $a_i \in \mathbb{R}^{d_2 \times \frac{H}{4} \times \frac{W}{4}}$  and then perform a two-step operation. First, the  $a_i$  is output to the outside for subsequent calculation of the auxiliary loss (channel cross-entropy loss). Then  $a_i$  and the original fusion feature  $B_i$  are residually connected. Finally, the channel enhanced feature  $B'_i \in \mathbb{R}^{d_2 \times \frac{H}{4} \times \frac{W}{4}}$  is obtained.

Figure 4c illustrates the structural diagram of the core module of CEB in Fig. 4a, known as the CTM. This structure can be further understood as being composed of two components:  $L$  iterations of transformer blocks and a final fixed receptive field convolutional layer. Figure 4b illustrates the inter-layer

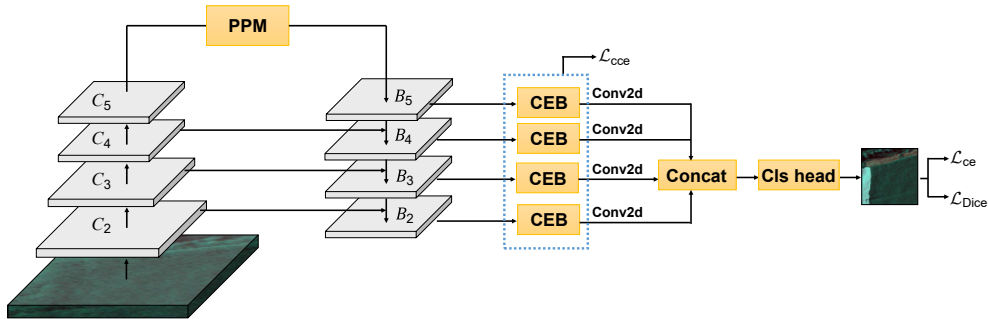
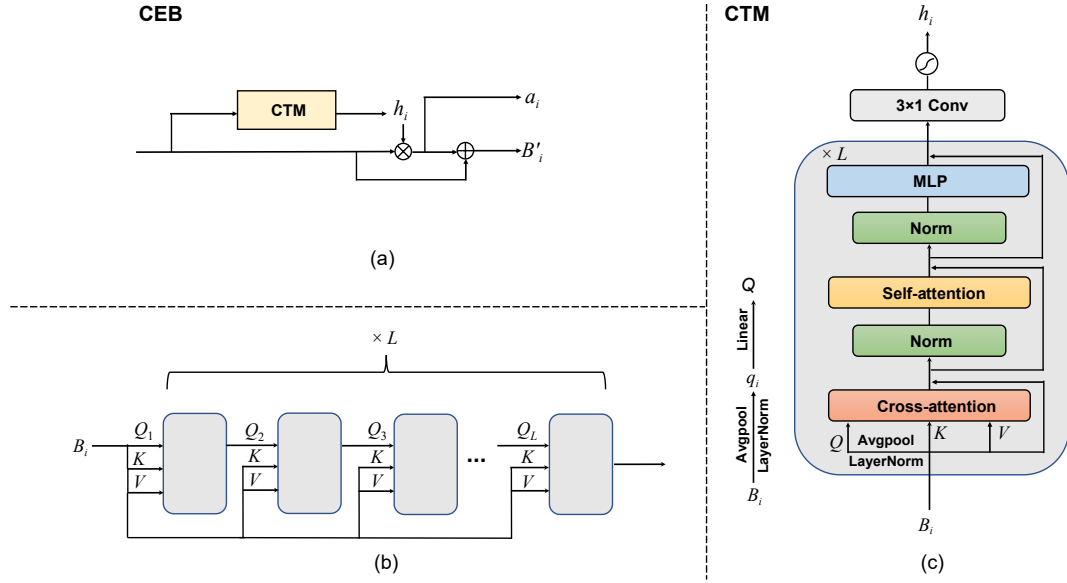


Fig. 3 Overall structure of CFormer. After the input image is extracted through the bottom-up encoder and top-down decoder of the semantic segmentation model, the model uses the features produced by each CEB to calculate the channel cross-entropy loss function ( $\mathcal{L}_{cce}$ ). In addition, the model uses the predicted image generated by the segmentation head (cls head) to calculate the cross-entropy loss function ( $\mathcal{L}_{ce}$ ) and the Dice loss function ( $\mathcal{L}_{Dice}$ ), respectively. The three loss functions work together to form a hybrid loss function, which improves the predictive ability of the model in each epoch of training. Conv2d denotes two-dimensional convolution.



**Fig. 4** Diagram of CEB and CTM structures, and the inter-layer transmission mechanism of the transformer. (a) Structure of CEB, the module can perform channel enhancement on the fusion feature  $B_i$ , and output the enhanced feature  $a_i$  before the residual connection to calculate the auxiliary loss. (b) Transfer relationship diagram between  $L$ -layer transformers.  $B_i$  as key ( $K$ ) and value ( $V$ ) will be re-used, and  $Q$  will be continuously updated. (c) Structure of CTM. The channel features are fully extracted through the transformer and the convolutional layer with a fixed field of view. On the left we show how  $B_i$  transforms into  $q_i$  and  $Q$  individually.

feature propagation within the  $L$  transformer blocks. We aim to hope that CTM can fully learn rich channel information from fusion feature  $B_i$ . The fusion feature  $B_i \in \mathbb{R}^{d_2 \times \frac{H}{4} \times \frac{W}{4}}$  will first go through a transformer composed of  $L$  layers, which is inspired by Ref. [47]. The initialization vector input  $q_i$  is obtained by performing global Average Pooling (Avgpool) and Layer Normalization (LayerNorm) on  $B_i$ , and the calculation formula is shown in Eq. (2),

$$q_i = \text{LayerNorm} \left( \frac{1}{H_i \times W_i} \sum_{h=1}^{H_i} \sum_{w=1}^{W_i} B_i(h, w) \right) \quad (2)$$

where  $q_i \in \mathbb{R}^{d_2 \times 1 \times 1}$ ,  $H_i \times W_i$  represents the scale of  $B_i$ , and  $B_i(h, w)$  represents the pixel located at  $(h, w)$  in  $B_i$ . The calculation formula of Cross-Attention (CA) used in CTM can be expressed as

$$\text{CA}(Q, K, V) = \text{Softmax} \left( \frac{QK^T}{\sqrt{C}} \right) V \quad (3)$$

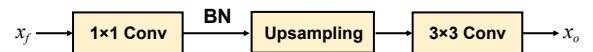
where  $Q \in \mathbb{R}^{d_2 \times 1 \times 1}$ , which is obtained from  $q_i$  through the linear layer, and  $K, V \in \mathbb{R}^{d_2 \times \frac{H}{4} \times \frac{W}{4}}$ , representing query, key, and value, respectively. The output of cross-attention will be input into self-attention after passing through LayerNorm. The calculation formula of Self-Attention (SA) in CTM can be expressed as

$$\text{SA}(\hat{Q}, \hat{K}, \hat{V}) = \text{Softmax} \left( \frac{\hat{Q}\hat{K}^T}{\sqrt{C}} \right) \hat{V} \quad (4)$$

where  $\hat{Q}, \hat{K}, \hat{V} \in \mathbb{R}^{d_2 \times 1 \times 1}$ , which represent the query, key, and value in self-attention respectively. After passing through the  $L$ -layer transformer, the transformer output is passed through a one-dimensional convolutional layer with a fixed field of view of 3 and a sigmoid layer. By using CTM, the transformer block can effectively capture the channel features in the fused feature  $B_i$ , further aggregate local features through a one-dimensional convolutional layer, and finally obtain the channel attention vector  $h_i \in \mathbb{R}^{d_2 \times 1 \times 1}$ .  $h_i$  will be used in CEB to perform channel enhancement on the fusion feature  $B_i$ .

The cls head of the main architecture is shown in Fig. 5. In Fig. 5,  $x_f$  is obtained by concatenating all the  $B'_i$  outputs from the CEB in Fig. 4c, and  $x_o$  represents the final output prediction image.

Finally, a pseudocode description of the CEB module



**Fig. 5** Cls head of the main structure in Fig. 3, which is different from the one of calculating channel cross-entropy loss ( $\mathcal{L}_{\text{cce}}$ ). The output of this block is denoted as  $x_o$ , which represents the final predicted segmentation map generated by the main model.

is presented in Algorithm 1. The input is feature map  $x$ . After applying global average pooling on it, the resulting vector is denoted as `channel_emb`. cablocks represent the  $L$ -layer transformer module, corresponding to Fig. 4a. By traversing all the cablocks, the `channel_emb` is continuously updated, and a fixed  $x$  is passed in each time. Transformer is good at capturing the global dependencies between `channel_emb` and feature maps, and modeling the contextual relationship between different channels inside `channel_emb`. The local channel information of `channel_emb` is then extracted using a one-dimensional convolution method  $f$  with a fixed field of view. `ca_feat` obtained by multiplying the feature map  $x$  and `channel_emb` represents  $a_i$  in Fig. 5c. The feature map is updated by adding, so far, the feature map  $x$  has completed the channel feature enhancement. Finally,  $x$  and `ca_feat` are returned at the same time for subsequent feature extraction and loss function calculation.

### 3.3 Hybrid loss

In CEB, the enhanced feature  $a_i$  is outputted in order to calculate an auxiliary loss function called channel cross-entropy loss ( $\mathcal{L}_{cce}$ ). It uses enhanced features to guide the model to learn the mapping to channel changes. The calculation process of  $\mathcal{L}_{cce}$  is shown in Fig. 6. In Fig. 3, each layer of CEB will output  $B'_i$  and enhanced features  $a_i \in \{a_2, a_3, a_4, a_5\}$  at the same time. Add all the enhanced features  $a_i$  for feature fusion, and then calculate the prediction image by interpolating a series of convolutions. This is different from the cls head in Fig. 5. In order to allow the auxiliary loss to be calculated more quickly using lightweight modules, a

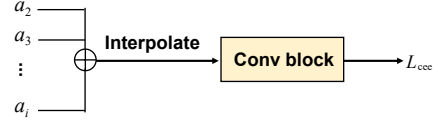
---

#### Algorithm 1 CEB module

---

**Require:** Input feature map  $x$

- 1: Compute channel embedding `channel_emb` by averaging pooling over  $x$ ;
  - 2: **for** each `attnBlock` in `cablocks` **do**
  - 3: `channel_emb`  $\leftarrow$  `attnBlock(channel_emb, x)`;
  - 4: **end for**
  - 5: Apply convolution layer  $f$  to `channel_emb`;
  - 6: Apply sigmoid function to `channel_emb`;
  - 7: Compute channel attention feature `ca_feat` =  $x \odot \text{channel\_emb}$  (where  $\odot$  denotes element-wise multiplication (Hadamard product));
  - 8: Update  $x \leftarrow x + \text{ca\_feat}$ ;
  - 9: **return** updated feature map  $x$  and channel attention feature `ca_feat`
- 



**Fig. 6** Calculation process of channel cross-entropy loss ( $\mathcal{L}_{cce}$ ).

one-dimensional convolutional layer is utilized to map features from high dimensions to the same dimension as the number of classes. By calculating the pixel-by-pixel cross entropy, we define  $\mathcal{L}_{cce}$  with Eq. (5),

$$\mathcal{L}_{cce}(x_a) = -\frac{1}{N_c} \sum_{c=1}^{C_c} \sum_{j=1}^{N_c} y_{jc} \ln g_{jc} \quad (5)$$

where  $N_c$  represents the number of pixels,  $C_c$  represents the semantic class contained in the enhanced feature prediction image,  $y_{jc}$  represents the real class of pixel  $j$ , if pixel  $j$  belongs to semantic class  $c$ , then  $y_{jc} = 1$ , otherwise  $y_{jc} = 0$ ,  $g_{jc}$  represents the probability that pixel  $j$  in the predicted image of enhanced features belongs to semantic class  $c$ . Similarly, a cross-entropy loss function is calculated as the main loss function for the prediction image calculated by the CFormer main frame, which is defined as Eq. (6),

$$\mathcal{L}_{ce}(x) = -\frac{1}{N_m} \sum_{c=1}^{C_m} \sum_{j=1}^{N_m} y_{jc} \ln p_{jc} \quad (6)$$

where  $N_m$  represents the number of pixels,  $C_m$  represents the semantic class contained in the prediction image,  $y_{jc}$  represents the true class of pixel  $j$ , and  $p_{jc}$  represents the probability that pixel  $j$  in the prediction map belongs to semantic class  $c$ .

Due to the serious class imbalance<sup>[48–50]</sup> in the agricultural aerial image semantic segmentation dataset, Dice loss<sup>[51]</sup> is introduced as an auxiliary loss function. Dice loss is a common loss function in semantic segmentation, which can directly improve Intersection over Union (IoU). It is defined as Eq. (7),

$$\mathcal{L}_{Dice}(x) = \frac{1}{C} \sum_{c=1}^C \left( 1 - \frac{2 \sum_{j=1}^N p_{jc} t_{jc}}{\sum_{j=1}^N p_{jc}^2 + \sum_{j=1}^N t_{jc}^2} \right) \quad (7)$$

where  $C$  represents the class contained in  $x$ ,  $p_{jc}$  represents the probability that the pixel  $j$  in the prediction image belongs to the class  $c$ , and  $t_{jc}$  represents the true value of the pixel  $j$  in the ground truth belonging to the class  $c$ . Combining the above three loss functions, the cross-entropy loss calculated

by the main frame is utilized as the main loss function, and use  $\mathcal{L}_{cce}$  and Dice loss as the auxiliary loss functions. Finally, the hybrid loss function we design can be expressed as Eq. (8),

$$\mathcal{L}_{total} = \frac{1}{1 + \alpha + \gamma} (1 \times \mathcal{L}_{ce} + \alpha \times \mathcal{L}_{cce} + \gamma \times \mathcal{L}_{Dice}) \quad (8)$$

In Eq. (8),  $\alpha$  and  $\gamma$  are used to control the weights of  $\mathcal{L}_{cce}$  and  $\mathcal{L}_{Dice}$ , respectively, and the sum of weights of all loss functions is controlled to be 1.

## 4 Experiment

### 4.1 Datasets and metrics

We follow the evaluation scheme described in Agriculture-Vision<sup>[4]</sup> and Tianchi suichang-round1<sup>[24]</sup> to evaluate the performance of our method, and conduct extensive experiments on the validation set. The training and testing are performed on the Agriculture-Vision-2021 dataset using both RGB images and RGBN images stitched with NIR images. But only RGBN images are trained and tested on Tianchi suichang-round1.

#### 4.1.1 Datasets

The Agriculture-Vision-2021 dataset is a large-scale multi-modal remote sensing segmentation dataset of aerial images under the agricultural model updated by Agriculture-Vision<sup>[4]</sup> in 2021. The dataset contains two–seven flight sequences in 54 fields of view from 2017 to 2020 with an image resolution of 10 cm. The dataset contains a total of 56 944 sets of high-resolution RGB images and NIR images in nine different categories. The validation set contains 18 334 sets of high-resolution Red, Green, and Blue (RGB) images and NIR images. Each image in the dataset has a fixed scale of  $512 \times 512$  pixels.

The Tianchi suichang-round1 dataset is provided by School of Environment and Resources, Zhejiang University, China to support data needs. This dataset is utilized for the Tianchi 2021 National Digital Ecology Innovation Competition. The data comprises high-resolution multispectral remote sensing images with resolutions ranging from 0.8 to 2 m. The imaging

bands include RGB and NIR bands. The dataset encompasses diverse landforms related to agriculture, including cultivated land, forests, and water bodies, as well as non-agricultural features like roads and urban land. The dataset consists of a total of 16 017 images, each with a size of  $256 \times 256$  pixels. Due to severe class imbalance and the absence of labeled validation set images provided by the official source, randomly selecting a proportion of the data for validation may lead to certain classes being completely absent in the validation set. Consequently, the entire dataset is employed as the validation set. For the training set, data augmentation is applied, and to address the imbalance, images from the minority classes are resampled to create a final training set of 34 323 images. Notably, data augmentation is exclusively applied to the training set, and the validation set remains unaltered.

An overview of the two datasets is presented in Table 1, revealing that Agriculture-Vision-2021 boasts a higher resolution and larger image scale in comparison to Tianchi suichang-round1. Both datasets are comprised of multi-modal remote sensing segmentation data with RGB and NIR bands. Notably, Agriculture-Vision-2021 is exclusively focused on agricultural scenes, distinguishing it as a purely agricultural dataset.

#### 4.1.2 Metrics

In remote sensing segmentation tasks, Mean Intersection over Union (MIoU) is a commonly used performance indicator to evaluate the segmentation accuracy of the model. MIoU is shown in Eq. (9),

$$\begin{aligned} \text{IoU}_k &= \frac{\text{TP}_k}{\text{TP}_k + \text{FP}_k + \text{FN}_k}, \\ \text{MIoU} &= \frac{1}{N} \sum_{k=1}^N \text{IoU}_k \end{aligned} \quad (9)$$

where  $N$  is the number of categories, and  $\text{TP}_k$  is the number of pixels in category  $k$  that are actually predicted to be positive.  $\text{FP}_k$  is the number of pixels in category  $k$  that are incorrectly predicted as positive examples and  $\text{FN}_k$  is the number of pixels in category  $k$

**Table 1** Relevant information of the two datasets Agriculture-Vision-2021 and Tianchi suichang-round1 involved in the experiments, where Agriculture-Vision-2021 is the dataset under the full agriculture scenario. In this table, “Training” indicates the number of images in the training set, and “Validation” indicates the number of images in the validation set.

| Dataset                 | Year | Training | Validation | Category | Resolution | Scale (pixel)    | Band     |
|-------------------------|------|----------|------------|----------|------------|------------------|----------|
| Agriculture-Vision-2021 | 2021 | 56 944   | 18 334     | 9        | 10 cm      | $512 \times 512$ | RGB, NIR |
| Tianchi suichang-round1 | 2021 | 34 323   | 16 017     | 10       | 0.8–2 m    | $256 \times 256$ | RGB, NIR |

that are incorrectly predicted as negative examples. IoU is a measure of segmentation accuracy by comparing the intersection and union between model predictions and ground truth labels.  $\text{IoU}_k$  of each category  $k$  is summed and averaged to obtain MIOU. MIOU is used to measure the segmentation effect of the model for each category. The value of MIOU ranges from 0 to 1, the closer the value is to 1, the better the segmentation performance of the model, and the lower the value, the poorer the segmentation effect of the model on some categories.

## 4.2 Implementation details

**Architecture.** Our semantic segmentation model uses IBN-Net101-a as the encoder and uses ImageNet to pretrain the encoder.

**Baselines.** For Agriculture-Vision-2021, our method is compared with several baselines in addition to several classical channel enhancement methods. The first validations are conducted on PSPNet<sup>[45]</sup>, FPN<sup>[46]</sup>, and DeepLabV3+<sup>[31]</sup>. These models all use ResNet-101 pretrained on ImageNet as the encoder. The IBN-Net101-a and IBN-Net101-s<sup>[21]</sup> are then verified for their effectiveness when applied on DeepLabV3+, drawing from previous research on Agriculture-Vision. Since the original optimization inspiration of our model comes from FPN, we verify the performance of FPN with IBN-Net101-a as the encoder. In addition, we also replace FPN with UPerNet<sup>[38]</sup> again to compare the performance between different decoders. We also verify the TFNet<sup>[21]</sup> of the channel attention module that also uses the transformer structure. Then we validate SE<sup>[5]</sup> and CBAM<sup>[6]</sup> by replacing the CTM with a traditional channel augmentation method. The encoders in the above methods are pre-trained on ImageNet. In addition to the above methods, We add SOTA models that use a transformer as the encoder, SegFormer using the official SegFormer-B5 as the pre-trained model, and TransUnet using R50-ViT-B as the pre-trained model in our experiments.

On the Tianchi suichang-round1 dataset, we also verify PSPNet<sup>[45]</sup>, FPN<sup>[46]</sup>, and DeepLabV3+<sup>[31]</sup>. In addition, we also verify TFNet<sup>[21]</sup>, which also uses the channel attention module of the transformer structure, and Agrisegnet<sup>[7]</sup>, which is applied in the field of agricultural multi-modal remote sensing segmentation. These models all use ResNet-101 as the encoder. Then we verify IBN-Net101-a and IBN-Net101-s using IBN-

Net101-a and IBN-s as the backbone, respectively. All the above encoders use ImageNet pre-training. Finally, we also verify SegFormer and TransUnet using a transformer as the encoder in the experiment.

**Training.** In order to maintain the environment consistency of the comparative experiments, on Agriculture-Vision-2021, all experiments use the same Intel Xeon Platinum 8255C CPU (2.50 GHz, 24 vCPUs), but only one NVIDIA GeForce RTX 3090 GPU (24 GB). In terms of datasets, we use random vertical flipping, horizontal flipping, and radial transformation data enhancement methods. For RGB image experiments, the image input scale is fixed as  $3 \times 512 \times 512$  pixels. For the RGBN image experiment, we fix the image input scale to  $4 \times 512 \times 512$  pixels by stitching together NIR images. We train baselines and our models using the SGD optimizer with a learning rate set to 0.01 and weight decay set to 0.0003 for 2000 warm-up iterations, 7000 static iterations, and 17 000 decay iterations. For SegFormer and TransUnet, the optimizer is changed to AdamW while training, leaving the other configurations unchanged.

For the Tianchi suichang-round1 dataset, all experiments use the same 24 vCPU Intel(R) Xeon(R) Platinum 8255C CPU @ 2.50 GHz, but only one 24 GB NVIDIA GeForce RTX 3090 GPU is used for training. In addition, AdamW is used as the optimizer, the learning rate is set to 0.0001, and the weight decay is set to 0.001. And no longer manually adjust the learning rate, but use PyTorch's cosine annealing method to adjust the learning rate. The total number of training epochs is set to 20. Data augmentation is used for the training set, while no adjustment is made for the validation set.

## 4.3 Results

### 4.3.1 RGB

On the Agriculture-Vision-2021 dataset, we conduct training and validation on RGB images. The results of our experiments on RGB images are shown in Table 2, which reflect the difficulty of this task. Among them, TFNet is the worst-performing method. Among the methods involved in many experiments, IBN-Net101-s is also based on the Agriculture-Vision design method. Compared with the ordinary DeepLabV3+, MIOU increases by 3.64%. While in our setting, its performance lags behind IBN-Net101-a by -1.7%. Since our decoder is inspired by FPN, we replace the decoder with FPN in the case of IBN-Net101-a to

**Table 2** Experiment results for training and testing using RGB images of the Agriculture-Vision-2021 dataset. In column names, from left to right, BG: Background, DP: Double Plant, DD: Drydown, ED: Endrow, ND: Nutrient Deficiency, PS: Planter Skip, WT: Water, WW: Waterways, WC: Weed Cluster. The best results are highlighted in bold.

(Unit: %)

| Method             | IoU          |              |              |              |              |              |              |              |              | MIoU         |
|--------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|                    | BG           | DP           | DD           | ED           | ND           | PS           | WT           | WW           | WC           |              |
| TFNet              | 69.42        | 4.84         | 12.57        | 0.22         | 2.79         | 11.19        | 54.41        | 6.54         | 2.62         | 18.29        |
| PSPNet             | 70.41        | 3.09         | 54.30        | 0            | 11.80        | 3.98         | 57.73        | 13.57        | 21.53        | 26.27        |
| TransUnet          | 58.33        | 20.30        | 61.18        | 17.62        | 35.24        | 28.53        | 52.88        | 18.91        | 31.54        | 36.06        |
| FPN                | 72.21        | 20.86        | 59.40        | 9.96         | 25.25        | 22.02        | 69.31        | 29.71        | 27.17        | 37.32        |
| DeepLabV3+         | 72.91        | 22.52        | 59.13        | 9.92         | 28.13        | 17.43        | 69.34        | 32.62        | 25.09        | 37.45        |
| IBN-Net101-s       | 73.29        | 26.14        | 60.93        | 19.99        | 30.82        | 29.82        | 66.56        | 32.76        | 29.46        | 41.09        |
| IBN-Net101-a (FPN) | 73.89        | 27.18        | 59.62        | 19.14        | 29.74        | 31.11        | 73.91        | 32.86        | 28.54        | 41.78        |
| IBN-Net101-a       | 73.64        | 27.96        | 60.82        | 22.09        | 33.45        | 32.22        | 69.93        | 34.01        | 31.01        | 42.79        |
| CFormer-SE         | 73.39        | 29.36        | 61.18        | 22.58        | 35.25        | 33.41        | 70.83        | 34.08        | 29.47        | 43.29        |
| CFormer-CBAM       | 73.56        | 28.76        | 61.25        | 22.27        | 33.75        | 32.15        | 70.88        | 35.34        | 29.94        | 43.10        |
| SegFormer          | <b>74.28</b> | 29.46        | 59.49        | 14.21        | 31.17        | <b>36.99</b> | <b>75.19</b> | <b>39.79</b> | 33.16        | 43.75        |
| CFormer (ours)     | 71.64        | <b>34.29</b> | <b>62.34</b> | <b>27.63</b> | <b>38.80</b> | 36.91        | 69.87        | 38.10        | <b>35.46</b> | <b>46.12</b> |

verify the differences between different decoders. From the experimental results, it can be observed that compared to the DeepLabV3+ decoder, the model's performance decreased by  $-1.01\%$ . Our method, CFormer, stands out among all compared methods. The IoU metric for almost all categories achieves the highest scores, surpassing the strong baseline IBN-Net101-s by  $5.03\%$ . In two sets of experiments, we replace the CTM in CFormer with SE and CBAM to compare the influence of different channel attention methods on the model. Finally, the performance of CFormer using CTM was improved compared to SE and CBAM by  $2.83\%$  and  $3.02\%$ , respectively. The results also include SegFormer and TransUnet of models using a transformer as the encoder to achieve SOTA performance in the segmentation field. It can be seen that CFormer, using a traditional convolutional network as the encoder, achieves better performance than the others.

#### 4.3.2 RGB-NIR

On RGBN images, we also conduct extensive experiments. The experimental results on the Agriculture-Vision-2021 dataset are shown in Table 3, and the experimental results on the Tianchi suichang-round1 dataset are shown in Table 4.

**Agriculture-Vision-2021.** Through the experimental results, we can intuitively see that the introduction of NIR images improves the performance of all baselines, which shows that NIR images have great application value for the model. The improvement that different

methods can bring to the model is also different. Among them, the performance of TFNet is still the worst, it improves MIoU by  $1.71\%$  after applying RGBN images. However, the overall performance of the TFNet is verified to be only  $20\%$ . For IBN-Net101-s, MIoU improves by  $1.74\%$  after applying RGBN images. After applying RGBN images to our method CFormer, it improves MIoU by  $2.71\%$ , and it is  $6\%$  higher than IBN-Net101-s. As in the experiments on RGB images, we replace the CTM in CFormer with SE and CBAM, and after applying RGBN images, the MIoU improvements are  $2.79\%$  and  $2.76\%$ , respectively. Finally, on multi-modal images, CFormer still outperforms SegFormer and TransUnet which use a transformer as the encoder.

In Table 5, we show the number of parameters and computational complexity required by the models involved in the experiments when training a  $24 \times 256 \times 256$  pixels image. Among them, because AgriSegnet uses a multi-scale training and prediction method, its computational complexity is far higher than other models, and it also brings a large number of parameters. For the models SegFormer and TransUnet that use a transformer as the encoder, due to the need for all features to participate in feature extraction during calculation, this series of models require a huge number of parameters, which directly leads to the need for more costs when these models are used. Compared with other methods, the computational complexity and number of parameters of CFormer are moderate, while

**Table 3** Experiments for training and testing using RGB images and NIR images of the Agriculture-Vision-2021 dataset. In column names, from left to right, BG: Background, DP: Double Plant, DD: Drydown, ED: Endrow, ND: Nutrient Deficiency, PS: Planter Skip, WT: Water, WW: Waterways, WC: Weed Cluster. The best results are highlighted in bold.

(Unit: %)

| Method                 | IoU          |              |              |              |              |              |              |              |              | MIoU         |
|------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|                        | BG           | DP           | DD           | ED           | ND           | PS           | WT           | WW           | WC           |              |
| TFNet                  | 68.86        | 6.38         | 14.62        | 1.77         | 2.58         | 13.16        | 57.90        | 11.10        | 3.66         | 20.00        |
| PSPNet                 | 71.52        | 1.96         | 53.92        | 0            | 25.63        | 8.50         | 61.40        | 20.32        | 18.74        | 29.11        |
| TransUnet              | 65.35        | 22.61        | 60.35        | 17.88        | 45.53        | 21.22        | 62.71        | 21.33        | 30.96        | 38.66        |
| FPN                    | 73.63        | 19.43        | 58.73        | 11.03        | 37.89        | 21.96        | 73.43        | 32.18        | 24.66        | 39.22        |
| DeepLabV3+             | 73.90        | 20.70        | 57.51        | 12.00        | 37.37        | 19.89        | 74.00        | 34.97        | 24.18        | 39.39        |
| IBN-Net101-s           | 74.39        | 25.46        | 60.12        | 19.34        | 42.84        | 27.22        | 73.06        | 34.92        | 28.12        | 42.83        |
| IBN-Net101-a (FPN)     | 75.35        | 26.34        | 58.48        | 17.58        | 40.94        | 31.28        | 77.83        | 35.94        | 28.27        | 43.56        |
| IBN-Net101-a (upernet) | 75.30        | 27.58        | 58.98        | 17.13        | 41.32        | 32.62        | <b>78.45</b> | 37.26        | 27.78        | 44.05        |
| IBN-Net101-a           | 75.14        | 30.11        | 59.79        | 23.64        | 44.45        | 33.67        | 73.25        | 39.75        | 30.41        | 45.58        |
| CFormer-SE             | 74.75        | 29.93        | 60.19        | 23.06        | 45.13        | 34.22        | 74.97        | 37.64        | 31.09        | 45.66        |
| CFormer-CBAM           | 75.41        | 29.71        | 59.98        | 22.92        | 44.05        | 33.64        | 76.23        | 37.98        | 31.26        | 45.69        |
| SegFormer              | <b>76.06</b> | 29.80        | 60.28        | 16.60        | 42.36        | <b>39.13</b> | 78.15        | <b>42.36</b> | 31.38        | 46.24        |
| CFormer (ours)         | 74.45        | <b>34.16</b> | <b>63.17</b> | <b>29.81</b> | <b>48.52</b> | 38.00        | 75.62        | 38.86        | <b>36.92</b> | <b>48.83</b> |

**Table 4** Verification results of each baseline model on Tianchi suichang-round1. In the table, FL: Farmland, FR: Forest, GR: grass, RD: Road, UA: Urban Area, CS: Countryside, IL: Industrial Land, CT: Construction, WT: Water, BR: Bare Ground. The best results are highlighted in bold.

(Unit: %)

| Model        | IoU          |              |              |              |              |              |              |              |              |              | MIoU         |
|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|              | FL           | FR           | GR           | RD           | UA           | CS           | IL           | CT           | WT           | BR           |              |
| PSPnet       | 65.01        | 90.65        | 2.09         | 21.03        | 18.69        | 53.96        | 30.30        | 12.91        | 66.53        | 1.27         | 36.24        |
| Agrisegnet   | 62.07        | 90.25        | 3.53         | 22.60        | 27.65        | 55.27        | 40.00        | 19.34        | 69.11        | 6.15         | 39.60        |
| TFNet        | 77.23        | 93.98        | 47.88        | 26.19        | 45.20        | 63.38        | 53.66        | 45.23        | 75.23        | 28.64        | 55.66        |
| FPN          | 79.35        | 94.51        | 47.65        | 28.86        | 52.06        | 65.97        | 57.06        | 46.20        | 77.95        | 27.33        | 57.69        |
| DeepLabV3+   | 79.22        | 94.52        | 47.92        | 30.98        | 54.65        | 67.28        | 58.49        | 48.57        | 78.20        | 31.70        | 59.15        |
| TransUnet    | 72.59        | 92.62        | 46.04        | 38.65        | 57.50        | 64.28        | 60.77        | 53.33        | 72.73        | 40.01        | 59.85        |
| IBN-Net101-s | 72.18        | 92.64        | 52.87        | 31.10        | 61.21        | 62.66        | 62.79        | 55.65        | 73.73        | 42.69        | 60.75        |
| SegFormer    | <b>83.33</b> | <b>95.54</b> | 61.05        | 36.56        | 55.93        | 72.23        | 58.22        | 59.07        | <b>81.58</b> | 34.88        | 63.84        |
| IBN-Net101-a | 82.16        | 95.27        | 61.75        | 37.34        | 63.06        | 71.58        | 66.72        | 60.82        | 80.80        | 43.11        | 66.26        |
| CFormer-CBAM | 82.45        | 95.36        | 62.37        | 38.34        | 63.23        | 72.15        | 66.35        | 61.16        | 80.78        | 44.36        | 66.66        |
| CFormer-SE   | 82.97        | 95.43        | 63.72        | <b>39.04</b> | 63.18        | <b>72.93</b> | 66.54        | 61.97        | 80.68        | 42.43        | 67.09        |
| CFormer      | 82.26        | 95.26        | <b>66.02</b> | 37.57        | <b>67.26</b> | 72.03        | <b>69.59</b> | <b>64.84</b> | 80.99        | <b>49.10</b> | <b>68.49</b> |

it maintains higher segmentation accuracy than many methods that also use a transformer as the encoder or a channel attention module, demonstrating the superiority of the proposed method.

**Tianchi suichang-round1.** On this dataset, we verify Agrisegetnet, which is based on multi-view training and is also designed in the agricultural domain, but its performance is only 0.34% higher than the performance of PSPNet. As a method with similar ideas to CFormer, TFNet also uses the channel

enhancement method of the transformer structure, which is different from the performance on Agriculture-Vision-2021, and its performance is 12.83% worse than CFormer. Furthermore, we also validate DeepLabV3+, IBN-Net101-s, and IBN-Net101-a. Among these, while ensuring the utilization of the same source code and experimental environment, IBN-Net101-s demonstrates consistent performance with the results on Agriculture-Vision-2021 and still trailed behind IBN-Net101-a, exhibiting

**Table 5** Number of parameters and computational complexity required by the models involved in the experiment to train a  $24 \times 256 \times 256$  pixels image (here,  $24 \times 256 \times 256$  denotes an input tensor with 24 channels and a spatial resolution of  $256 \times 256$  pixels). In the table, # denotes the index number, and all entries in the subsequent table convey this meaning.  $M = 10^6$  and  $G = 10^9$ . Bold numbers indicate the best results.

| Index number | Method       | Number of parameters (M) | FLOPs (G)    |
|--------------|--------------|--------------------------|--------------|
| #1           | AgriseNet    | 59.06                    | <b>92.62</b> |
| #2           | TFNet        | 48.20                    | 14.79        |
| #3           | FPN          | 45.11                    | 12.78        |
| #4           | DeepLabV3+   | 45.68                    | 14.17        |
| #5           | TransUnet    | <b>93.23</b>             | 32.31        |
| #6           | IBN-Net101-s | 59.63                    | 23.23        |
| #7           | SegFormer    | 84.60                    | 24.97        |
| #8           | IBN-Net101-a | 59.64                    | 23.24        |
| #9           | CFormer      | 69.60                    | 35.59        |

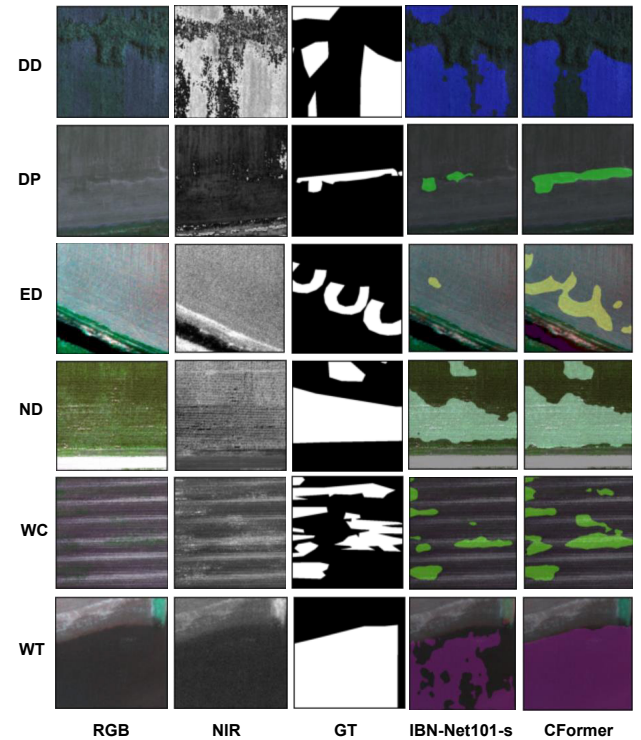
a performance difference of 5.51%. In conclusion, we validate the proposed CFormer in this paper. Compared to other methods, AgriseNet and IBN-Net101-s, which are also based on multi-modal remote sensing segmentation tasks in the agricultural domain, CFormer demonstrates exceptionally strong performance, with performance improvements of 28.89% and 7.74%, respectively. Similarly, on this dataset, the performance of CFormer is still better than that of SegFormer and TransUnet using a transformer as the encoder, which is 4.65% and 8.64% higher than that of SegFormer and TransUnet on MIoU.

Based on the experimental results from Tables 3 and 4, it is evident that, under the premise of utilizing the source code, the performance of TFNet, another channel enhancement method employing a transformer structure, exhibits significant variations across different datasets. Agriculture-Vision-2021, in contrast to Tianchi suichang-round1, boasts higher resolution and larger image scales, thus imposing greater training challenges on the dataset. Coupled with the specialized context of agricultural multi-modal scenarios, TFNet encounters substantial challenges in feature extraction on such demanding datasets. In contrast to TFNet, our proposed CFormer demonstrates robust segmentation capabilities across both agricultural and general remote sensing scenarios. This can be attributed to the CTM, which optimally leverages channel features to enhance fusion features, and the well-calibrated Dual-CE-Dice loss.

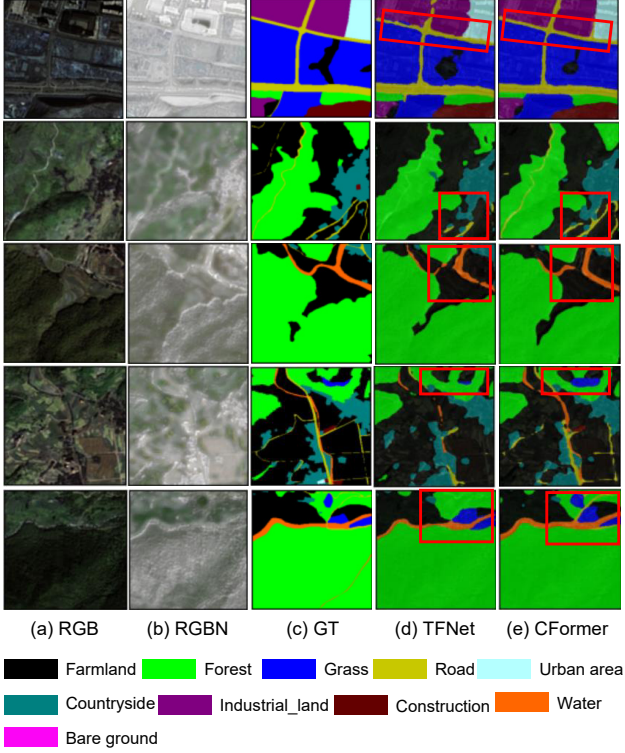
Finally, we visualize the segmentation results of the model on different datasets. We visualize IBN-Net101-s and CFormer on the Agriculture-Vision-2021 dataset and present the results in Fig. 7. Furthermore, we visualize TFNet and CFormer in the Tianchi suichang-round1 dataset and present the results in Fig. 8. From Figs. 7 and 8, it can be observed that, in comparison to other models, CFormer demonstrates the ability to more accurately and intricately delineate pixels belonging to different classes.

## 5 Ablation Study

**Agriculture-Vision-2021.** In this section, we conduct a large number of ablation experiments to verify the effectiveness of our method. It mainly involves the following four aspects: (1) Our improved basic decoder (BD), which is used for the subsequent introduction of the transformer module, (2) the introduction of our CEB, (3) the variable  $\alpha$  in our designed Dual-CE-Dice



**Fig. 7** Visualization of model predictions on the Agriculture-Vision-2021 validation set. Rows from top to bottom display different agricultural anomaly categories: DD: Drydown, DP: Double Plant, ED: Endrow, ND: Nutrient Deficiency, WC: Weed Cluster, and WT: Water. Columns from left to right present the input RGB (color) and NIR images, the Ground Truth (GT) mask, the predictions from the baseline model IBN-Net101-s, and the predictions from our proposed CFormer model, respectively.



**Fig. 8** Visualization results of several well-trained models on the Tianchi suichang-round1 dataset are presented. The subfigures showcase: (a) RGB images, (b) RGBN images, (c) Ground Truth (GT), as well as prediction outcomes for the same image by TFNet and CFormer. The content represented by each color block is labeled below the respective image. Notably, specific details in the images are highlighted with red boxes, revealing that CFormer achieves finer and more accurate segmentation results compared to TFNet.

loss is used to control the weight of  $\mathcal{L}_{cce}$  in the loss function, and (4) the variable  $\gamma$  is used to control the weight of  $\mathcal{L}_{Dice}$  in the Dual-CE-Dice loss, and the experimental results are shown in Table 6.

First, we verify the performance of the model inspired by FPN to optimize the DeepLabV3+ decoder, as the basis for applying CEB. Then we introduce CEB to it and verify the performance. Since the weights  $\alpha$  and  $\gamma$  of two auxiliary loss functions are involved in our designed Dual-CE-Dice loss, we explore the performance of the model under different parameter settings respectively. All the above ablation experiments use RGBN images for training and validation. From the results in Table 6, we can see that our optimization of the DeepLabV3+ decoder brings 0.24% performance improvement (Table 6, #1 vs #2). Combining the CFormer with SE and CBAM applied in Table 3, the experimental results verify our conjecture that the computationally simple channel enhancement method is not suitable for the fusion features of the decoder, which instead brings a negative impact on the model performance. Compared with our BD (Table 6, #2), SE and CBAM degrade the performance by 0.16% and 0.13%. In contrast, the introduction of CEB improves the model performance by 0.32%. Finally, under the joint effect of  $\alpha = 0.5$  and  $\gamma = 1$ , the performance of the final model is improved by 3.25% compared with the baseline IBN-Net101-a. In addition, we also verify the number of parameters and model performance of CTM with different numbers of layers ( $L$ ), and the results are shown in Table 7.

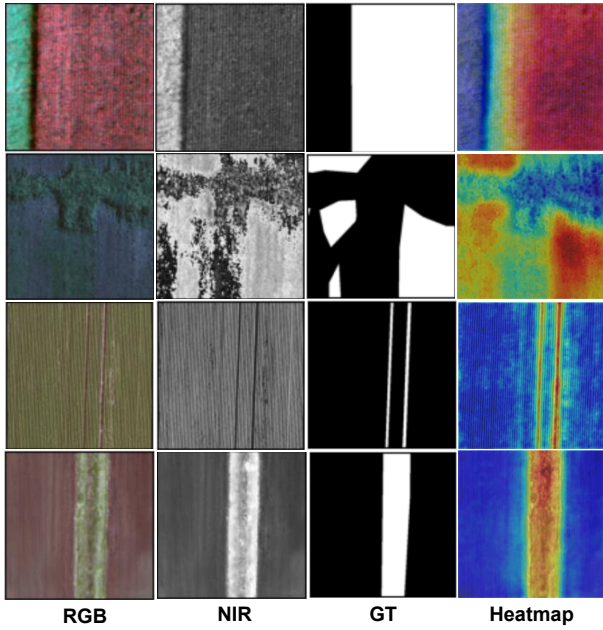
As CTM fundamentally operates as an attention mechanism, we also visualize the attention as heatmaps. In Fig. 9, we utilize the enhanced features generated by CEB to create heatmaps, while simultaneously displaying RGB images, NIR images, and the ground truth. From Fig. 9, it is evident that the enhanced features in the heatmap concentrate attention

**Table 6** Experiments for ablation studies using RGBN images of the Agriculture-Vision-2021 dataset. In column names, from left to right, BD: Basic Decoder, CEB: Channel Enhancement Block,  $\alpha$ : weight controller for  $\mathcal{L}_{cce}$  in Dual-CE-Dice loss,  $\gamma$ : weight controller for  $\mathcal{L}_{dice}$  in Dual-CE-Dice loss, BG: Background, DP: Double Plant, DD: Drydown, ED: Endrow, ND: Nutrient Deficiency, PS: Planter Skip, WT: Water, WW: Waterways, WC: Weed Cluster. In the table, the best results are highlighted in bold,  $\checkmark$  denotes that the module is enabled, and  $\times$  denotes that it is disabled.

| Index number | BD           | CEB          | $\alpha$ | $\gamma$ | IoU (%)      |              |              |              |              |              |              |              |              | MIoU (%)     |
|--------------|--------------|--------------|----------|----------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|              |              |              |          |          | BG           | DP           | DD           | ED           | ND           | PS           | WT           | WW           | WC           |              |
| #1           | $\times$     | $\times$     | 0        | 0        | 75.14        | 30.11        | 59.79        | 23.64        | 44.45        | 33.67        | 73.25        | 39.75        | 30.41        | 45.58        |
| #2           | $\checkmark$ | $\times$     | 0        | 0        | 75.26        | 31.04        | 60.57        | 24.55        | 43.36        | 31.89        | 75.28        | 41.03        | 29.41        | 45.82        |
| #3           | $\checkmark$ | $\checkmark$ | 0        | 0        | 75.09        | 31.91        | 59.28        | 24.89        | 45.41        | 33.46        | <b>76.98</b> | 37.47        | 30.74        | 46.14        |
| #4           | $\checkmark$ | $\checkmark$ | 1.0      | 0        | <b>75.38</b> | 30.54        | 59.49        | 25.75        | 45.96        | 34.60        | 74.99        | <b>41.32</b> | 30.80        | 46.54        |
| #5           | $\checkmark$ | $\checkmark$ | 0        | 1.0      | 73.79        | 33.00        | 61.94        | 28.10        | 47.53        | 33.98        | 76.78        | 40.08        | 34.70        | 47.77        |
| #6           | $\checkmark$ | $\checkmark$ | 1.0      | 1.0      | 74.44        | <b>34.87</b> | 62.87        | 28.67        | <b>48.67</b> | 37.42        | 74.25        | 38.37        | 35.61        | 48.35        |
| #7           | $\checkmark$ | $\checkmark$ | 0.5      | 1.0      | 74.45        | 34.16        | <b>63.17</b> | <b>29.81</b> | 48.52        | <b>38.00</b> | 75.62        | 38.86        | <b>36.92</b> | <b>48.83</b> |

**Table 7** Experiments of model performance with different CTM layers ( $L$ ), which show that CTM reaches the best when  $L = 5$ .

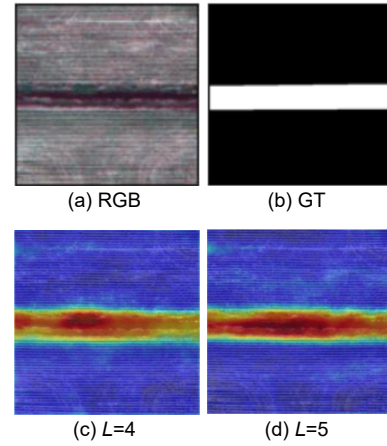
| Index number | $L$ | Number of parameters (M) | MIoU (%) |
|--------------|-----|--------------------------|----------|
| #1           | 1   | 54                       | 47.42    |
| #2           | 2   | 58                       | 48.11    |
| #3           | 3   | 62                       | 48.25    |
| #4           | 4   | 65                       | 47.95    |
| #5           | 5   | 69                       | 48.83    |
| #6           | 6   | 73                       | 48.42    |



**Fig. 9** Visualization of model attention via CEB enhanced features. We utilize the enhanced features generated by CEB as attention to visualize heatmaps, demonstrating where the model's focus lies. In the sub figures, we present RGB images, NIR images, GT, and heatmaps, respectively. All images are sourced from Agriculture-Vision-2021.

on pixels belonging to the correct class, effectively aiding the model in accomplishing remote sensing segmentation tasks. Additionally, we visualize the impact of varying the value of  $L$  on CTM across different situations and present the results in Fig. 10. Notably, when  $L$  is set to 5, compared to the case when  $L$  is 4, the attention spans more broadly and converges more strongly within the scope of the ground truth. Through the heatmaps in Figs. 9 and 10, we visually validate the efficacy of our proposed transformer-based CTM module.

**Tianchi suichang-round1.** We also conduct extensive ablation studies on this dataset, and the



**Fig. 10** Visualization of enhanced features with different  $L$  values in CTM. In CTM,  $L$  is an adjustable parameter. To provide a clearer interpretation of the results from ablation studies, we also visualize heatmaps of enhanced features for different values of  $L$ . As shown in the figure, (a) represents the RGB image, (b) shows the GT, and (c) and (d) present heatmaps with  $L$  set to 4 and 5, respectively. All images are sourced from Agriculture-Vision-2021.

results are similar to those in Table 5. Our BD improves the performance by 0.51% compared with the baseline (Table 8, #1 vs #2). After introducing CEB, the performance improves by another 0.22%. The difference from Table 6 is the selection of adjustable parameters  $\alpha$  and  $\gamma$  in the Dual-CE-Dice loss. Through experimentation, we find that when  $\gamma$  is set to a small value, the model has stronger learning ability. Finally, when  $\alpha$  is set to 1 and  $\gamma$  is set to 0.1, the model performance improves by 2.23% compared with the baseline. Similarly, in Table 4, we replace CTM with SE and CBAM to verify the performance of different methods. Compared with the ablation study of the BD (#2 in Table 8), CBAM once again has a negative impact on the model, with a performance decrease of 0.11%. On this relatively easy dataset, SE improves the performance by 0.12% compared with BD.

It is worth mentioning that the channel information enhancement method is not a new research direction. In the past, a large number of studies used channel attention to enhance channel information. These methods are mostly applied in encoders to improve performance. But here comes a question, if the channel features in the encoder can be enhanced, why not the fusion features in the decoder? The fusion features in the agricultural multi-modal aerial image semantic segmentation task contain not only spatial and semantic information but also additional multi-modal

**Table 8** Experiments for ablation studies using RGBN images of the Tianchi suichang-round1 dataset. In column names, from left to right, BD: Basic Decoder, CEB: Channel Enhancement Block,  $\alpha$ : weight controller for  $L_{\text{ce}}$  in Dual-CE-Dice loss,  $\gamma$ : weight controller for  $L_{\text{dice}}$  in Dual-CE-Dice loss, farmland: FL, forest: FR, grass: GR, road: RD, urban area: UA, countryside: CS, industrial land: IL, construction: CT, water: WT, bare ground: BR. In the table, the best results are highlighted in bold,  $\checkmark$  denotes that the module is enabled, and  $\times$  denotes that it is disabled.

| Index number | BD           | CEB          | $\alpha$ | $\gamma$ | IoU (%)      |              |              |              |              |              |              |              |              |              | MIoU (%)     |
|--------------|--------------|--------------|----------|----------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|              |              |              |          |          | FL           | FR           | GR           | RD           | UA           | CS           | IL           | CT           | WT           | BR           |              |
| #1           | $\times$     | $\times$     | 0        | 0        | 82.16        | 95.27        | 61.75        | 37.34        | 63.06        | 71.58        | 66.72        | 60.82        | 80.80        | 43.11        | 66.26        |
| #2           | $\checkmark$ | $\times$     | 0        | 0        | 82.74        | 95.39        | 62.38        | 37.86        | 63.96        | 72.67        | 67.30        | 61.13        | 80.53        | 43.76        | 66.77        |
| #3           | $\checkmark$ | $\checkmark$ | 0        | 0        | 82.60        | 95.40        | 62.74        | 38.89        | 64.21        | 72.31        | 67.22        | 61.54        | 81.03        | 43.93        | 66.99        |
| #4           | $\checkmark$ | $\checkmark$ | 1        | 0        | <b>83.10</b> | <b>95.50</b> | 64.85        | 39.79        | 66.45        | <b>73.29</b> | 67.99        | 63.58        | <b>81.45</b> | 44.10        | 68.01        |
| #5           | $\checkmark$ | $\checkmark$ | 0        | 0.10     | 81.61        | 95.09        | 64.87        | 38.83        | 66.89        | 71.33        | 68.75        | 63.61        | 80.60        | 48.82        | 68.04        |
| #6           | $\checkmark$ | $\checkmark$ | 1        | 0.40     | 80.99        | 94.87        | 64.73        | <b>41.04</b> | 65.99        | 71.17        | 69.27        | 63.82        | 79.75        | 47.92        | 67.95        |
| #7           | $\checkmark$ | $\checkmark$ | 1        | 0.10     | 82.26        | 95.26        | <b>66.02</b> | 37.57        | <b>67.26</b> | 72.03        | <b>69.59</b> | <b>64.84</b> | 80.99        | <b>49.10</b> | <b>68.49</b> |

information. We initially thought that these encoder-based channel information enhancement methods may not be suitable for fusion features containing complex information, which is validated in the experiments of this paper. The experiments involve two classic and powerful channel attention methods, SE and CBAM. However, the results suggest that when applied to the fusion features in the decoder, they can sometimes weaken the model on different datasets. In contrast, the previous channel information enhancement methods often consist of simple computational steps and can enhance the feature extraction capability of the encoder while consuming less computational cost. However, applying these methods to the fusion features in the decoder may limit the model’s learning ability. Therefore, we design a transformer-based CEB and Dual-CE-Dice loss to help CEB train better and effectively improve the performance of the model. Through the ablation studies in Tables 6 and 8, we find that both BD, CEB, and Dual-CE-Dice loss have strong robustness on different datasets.

**Class imbalance.** Our Dual-CE-Dice loss improves class imbalances. In the relevant literature of frameworks such as DeepLabV3+[31], FPN[46], and channel enhancement methods such as SENet[5] and CBAM[6], the problem of class imbalance is not mentioned. Therefore, the default loss function of the above method is the pixel-level cross-entropy loss function. The means we use to alleviate class imbalance is to introduce Dice loss in the Dual-CE-Dice loss by weighting. In ablation experiments #3 and #5 in Table 5,  $\gamma$  represents the weight of Dice loss. It can be clearly seen that the IoU of almost all classes in #5 is improved, which is more balanced than that of #3

without alleviating class imbalance. A similar situation occurred in another set of experiments, as shown in #3 and #5 in Table 8. On the whole, the introduction of the loss function to alleviate class imbalance has greatly improved the performance of the model. In Tables 6 and 8, the introduction of  $\gamma$  improves the performance by 1.63% and 1.05%, respectively, as shown in #3 and #5. It can be seen that class imbalance is a problem that cannot be ignored in the semantic segmentation task of agricultural multi-modal aerial images.

## 6 Conclusion

In the semantic segmentation task of multi-modal aerial images in agricultural scenarios, channel information enhancement methods are crucial for exploiting the potential of multi-modal data. Previous works hardly noticed the mining value of fusion features in the decoder, as fusion features often contain a large amount of features, including up-sampled spatial features, downsampled semantic features, and multi-modal features. If the fusion features are not further processed, it may lead to feature redundancy and other issues. However, blindly applying existing methods such as SE and CBAM to enhance the fusion features may have the opposite effect. In addition, the class imbalance problem in this task cannot be ignored.

To address these challenges, we introduce a CEB based on transformers into a trainable end-to-end framework called CFormer. CEB helps to directly extract and enhance channel features from the decoder’s fusion features. Moreover, we propose Dual-CE-Dice loss that not only helps the model effectively learn CEB but also alleviates the impact of class imbalance. Extensive comparative experiments are

conducted on the Agriculture-Vision-2021 and Tianchi suichang-round1 datasets, and the proposed method CFormer is proven to significantly outperform previous methods. The individual modules of CFormer have also been extensively ablated to demonstrate their strong robustness on different datasets.

### Acknowledgment

The work was supported partly by Guangxi Natural Science Foundation (No. 2025GXNSFFA069015), the National Natural Science Foundation of China (Nos. 62372119 and 62166003), the Guangxi Bagui Young Talent Training Program, the Innovation Project of Guangxi Graduate Education (Nos. JGY2024080 and JGY2022046), and the Guangxi Collaborative Innovation Center of Multi-Source Information Integration, China.

### Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

### References

- [1] Q. Li and Q. Zhao, Weakly-supervised semantic segmentation of airborne LiDAR point clouds in Hong Kong urban areas, in *Proc. 2023 Joint Urban Remote Sensing Event*, Heraklion, Greece, 2023, pp. 1–4.
- [2] T. Chowdhury and M. Rahnemoonfar, Attention based semantic segmentation on UAV dataset for natural disaster damage assessment, in *Proc. 2021 IEEE Int. Geoscience and Remote Sensing Symp.*, Brussels, Belgium, 2021, pp. 2325–2328.
- [3] M. Vignesh and Uma Priyadarsini P. S, Multispectral satellite image segmentation of land cover area using novel naive Bayes in comparison with decision tree, in *Proc. 8<sup>th</sup> Int. Conf. Science Technology Engineering and Mathematics*, Chennai, India, 2023, pp. 1–5.
- [4] M. T. Chiu, X. Xu, Y. Wei, Z. Huang, A. G. Schwing, R. Brunner, H. Khachatrian, H. Karapetyan, I. Dozier, G. Rose, et al., Agriculture-vision: A large aerial image database for agricultural pattern analysis, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Seattle, WA, USA, 2020, pp. 2825–2835.
- [5] J. Hu, L. Shen, and G. Sun, Squeeze-and-excitation networks, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 7132–7141.
- [6] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, CBAM: Convolutional block attention module, in *Proc. 15<sup>th</sup> European Conf. Computer Vision*, Munich, Germany, 2018, pp. 3–19.
- [7] M. U. Yaseen, M. M. Nasralla, F. Aslam, S. S. Ali, and S. B. A. Khattak, A novel approach based on multi-level bottleneck attention modules using self-guided dropblock for person re-identification, *IEEE Access*, vol. 10, pp. 123160–123176, 2022.
- [8] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, ECA-Net: Efficient channel attention for deep convolutional neural networks, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Seattle, WA, USA, 2020, pp. 11531–11539.
- [9] H. Sheng, R. Cong, D. Yang, R. Chen, S. Wang, and Z. Cui, UrbanLF: A comprehensive light field dataset for semantic segmentation of urban scenes, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 11, pp. 7880–7893, 2022.
- [10] Y. Wang, G. Li, and Z. Liu, SGFNet: Semantic-guided fusion network for RGB-thermal semantic segmentation, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7737–7748, 2023.
- [11] T. Panboonyuen, K. Jitkajornwanich, S. Lawawirojwong, P. Srestasathien, and P. Vateekul, Semantic segmentation on remotely sensed images using an enhanced global convolutional network with channel attention and domain specific transfer learning, *Remote Sens.*, vol. 11, no. 1, p. 83, 2019.
- [12] L. Ding, H. Tang, and L. Bruzzone, LANet: Local attention embedding to improve the semantic segmentation of remote sensing images, *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 426–435, 2021.
- [13] S. Huang, Z. Lu, R. Cheng, and C. He, FaPN: Feature-aligned pyramid network for dense image prediction, in *Proc. IEEE/CVF Int. Conf. Computer Vision*, Montreal, Canada, 2021, pp. 844–853.
- [14] J. Geng, S. Song, and W. Jiang, Dual-path feature aware network for remote sensing image semantic segmentation, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 5, pp. 3674–3686, 2024.
- [15] G. Li, Y. Wang, Z. Liu, X. Zhang, and D. Zeng, RGB-T semantic segmentation with location, activation, and sharpening, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 3, pp. 1223–1235, 2023.
- [16] H. Li, D. H. Zhai, and Y. Xia, ERDUnet: An efficient residual double-coding Unet for medical image segmentation, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2083–2096, 2024.
- [17] H. Li, K. Qiu, L. Chen, X. Mei, L. Hong, and C. Tao, SCAttNet: Semantic segmentation network with spatial and channel attention mechanism for high-resolution remote sensing images, *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 5, pp. 905–909, 2021.
- [18] C. Peng, T. Tian, C. Chen, X. Guo, and J. Ma, Bilateral attention decoder: A lightweight decoder for real-time semantic segmentation, *Neural Netw.*, vol. 137, pp. 188–199, 2021.
- [19] T. L. B. Khanh, D. P. Dao, N. H. Ho, H. J. Yang, E. T. Baek, G. Lee, S. H. Kim, and S. B. Yoo, Enhancing U-Net with spatial-channel attention gate for abnormal tissue segmentation in medical imaging, *Appl. Sci.*, vol. 10, no. 17, p. 5729, 2020.

- [20] C. Peng and J. Ma, Semantic segmentation using stride spatial pyramid pooling and dual attention decoder, *Pattern Recognit.*, vol. 107, p. 107498, 2020.
- [21] T. Wang, Z. Lai, and H. Kong, TFNet: Transformer fusion network for ultrasound image segmentation, in *Proc. 6<sup>th</sup> Asian Conf. Pattern Recognition*, Jeju Island, Republic of Korea, 2021, pp. 314–325.
- [22] A. Tavera, F. Cermelli, C. Masone, and B. Caputo, Pixel-by-pixel cross-domain alignment for few-shot semantic segmentation, in *Proc. IEEE/CVF Winter Conf. Applications of Computer Vision*, Waikoloa, HI, USA, 2022, pp. 1959–1968.
- [23] X. Yuan, X. Xu, X. Wang, K. Zhang, L. Liao, Z. Wang, and C. W. Lin, OSAP-Loss: Efficient optimization of average precision via involving samples after positive ones towards remote sensing image retrieval, *CAAI Trans. Intell. Technol.*, vol. 8, no. 4, pp. 1191–1212, 2023.
- [24] Tianchi, Suichang\_round1 dataset, <https://aistudio.baidu.com/datasetdetail/70670>, 2021.
- [25] J. Long, E. Shelhamer, and T. Darrell, Fully convolutional networks for semantic segmentation, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 3431–3440.
- [26] S. Gidaris and N. Komodakis, Object detection via a multi-region and semantic segmentation-aware CNN model, in *Proc. IEEE Int. Conf. Computer Vision*, Santiago, Chile, 2015, pp. 1134–1142.
- [27] W. Su and Z. Wang, Widening residual skipped network for semantic segmentation, *IET Image Process.*, vol. 11, no. 10, pp. 880–887, 2017.
- [28] F. Shen, Z. H. Wang, and Z. M. Lu, Weakly supervised classification model for zero-shot semantic segmentation, *Electron. Lett.*, vol. 56, no. 23, pp. 1247–1250, 2020.
- [29] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, DeepLab: Semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2018.
- [30] L. Gong, Y. Zhang, Y. Zhang, Y. Yang, and W. Xu, Erroneous pixel prediction for semantic image segmentation, *Comput. Visual Media*, vol. 8, no. 1, pp. 165–175, 2022.
- [31] Y. Chen, D. Wang, Z. Chen, Z. X. Yang, and E. Wu, Global video object segmentation with spatial constraint module, *Comput. Visual Media*, vol. 9, no. 2, pp. 385–400, 2023.
- [32] X. Pan, P. Luo, J. Shi, and X. Tang, Two at once: Enhancing learning and generalization capacities via IBN-Net, in *Proc. 15<sup>th</sup> European Conf. Computer Vision*, Munich, Germany, 2018, pp. 484–500.
- [33] D. Ulyanov, A. Vedaldi, and V. Lempitsky, Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 4105–4113.
- [34] S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in *Proc. 32<sup>nd</sup> Int. Conf. Machine Learning*, Lille, France, 2015, pp. 448–456.
- [35] S. Yang, S. Yu, B. Zhao, and Y. Wang, Reducing the feature divergence of RGB and near-infrared images using Switchable Normalization, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops*, Seattle, WA, USA, 2020, pp. 206–211.
- [36] D. Neupane, Y. Kim, and J. Seok, Bearing fault detection using scalogram and switchable normalization-based CNN (SN-CNN), *IEEE Access*, vol. 9, pp. 88151–88166, 2021.
- [37] Q. Liu, M. Kampffmeyer, R. Jenssen, and A. B. Salberg, Multi-view self-constructing graph convolutional networks with adaptive class weighting loss for semantic segmentation, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops*, Seattle, WA, USA, 2020, pp. 199–205.
- [38] T. Anand, S. Sinha, M. Mandal, V. Chamola, and F. R. Yu, AgriSegNet: Deep aerial semantic segmentation framework for IoT-assisted precision agriculture, *IEEE Sensors J.*, vol. 21, no. 16, pp. 17581–17590, 2021.
- [39] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, SegFormer: Simple and efficient design for semantic segmentation with transformers, in *Proc. 35<sup>th</sup> Int. Conf. Neural Information Processing Systems*, Virtual Event, 2021, p. 924.
- [40] J. Chen, J. Mei, X. Li, Y. Lu, Q. Yu, Q. Wei, X. Luo, Y. Xie, E. Adeli, Y. Wang, et al., TransUNet: Rethinking the U-Net architecture design for medical image segmentation through the lens of transformers, *Med. Image Anal.*, vol. 97, p. 103280, 2024.
- [41] X. Li, W. Wang, X. Hu, and J. Yang, Selective kernel networks, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019, pp. 510–519.
- [42] J. Fu, J. Liu, H. Tian, Yong Li, Y. Bao, Z. Fang, and H. Lu, Dual attention network for scene segmentation, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019, pp. 3141–3149.
- [43] Z. Wei, D. Liang, D. Zhang, L. Zhang, Q. Geng, M. Wei, and H. Zhou, Learning calibrated-guidance for object detection in aerial images, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 15, pp. 2721–2733, 2022.
- [44] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 770–778.
- [45] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, Pyramid scene parsing network, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 6230–6239.
- [46] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, Feature pyramid networks for object detection, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 936–944.

- [47] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, Attention is all you need, in *Proc. 31<sup>st</sup> Int. Conf. Neural Information Processing Systems*, Long Beach, CA, USA, 2017, pp. 6000–6010.
- [48] S. Tang, S. Huang, C. Zheng, E. Liu, C. Zong, and Y. Ding, A novel cross-project software defect prediction algorithm based on transfer learning, *Tsinghua Science and Technology*, vol. 27, no. 1, pp. 41–57, 2022.
- [49] Lou X, Li X, Meng H, P. Li, L. Zhong, and Z. Li. EEG-GANet: Simulating electroencephalogram data to address sample imbalance in P300 speller. *Tsinghua Science and Technology*, DOI: 10.26599/TST.2025.9010083.
- [50] Z. Cui, Z. Gao, S. Yue, R. Wang, and H. Zhu, Classification hardness based adaptive sampling ensemble for imbalanced data classification, *Tsinghua Science and Technology*, vol. 30, no. 6, pp. 2419–2433, 2025.
- [51] F. Milletari, N. Navab, and S. A. Ahmadi, V-Net: Fully convolutional neural networks for volumetric medical image segmentation, in *Proc. 4<sup>th</sup> Int. Conf. 3D Vision*, Stanford, CA, USA, 2016, pp. 565–571.



**Guangquan Lu** received the BEng degree in computer science from Guangxi Normal University, China, in 2011, and the PhD degree in philosophy from Sun Yat-sen University, China, in 2019. He is currently an associate professor with the Key Lab of Education Blockchain and Intelligent Technology, Ministry of Education, Guangxi Normal University, China. His research interests include deep learning, data mining, and graph representation learning for agricultural machine learning.



**Shilong Lin** received the BEng degree in software engineering from the Institute of Technology, East China Jiaotong University, China, in 2023. He is currently pursuing the master degree at Guangxi Normal University, China. His research interests include deep learning and graph representation learning.



**Shichao Zhang** received the PhD degree from Deakin University, Australia. He is currently a professor at Guangxi Normal University, China. He has published 90 international journal papers and over 70 international conference papers. His research interests include data mining and big data.



**Jiacheng Jiang** received the BEng degree in electronic information engineering from Zhejiang University of Science and Technology, China, in 2017. He is currently pursuing the master degree at Guangxi Normal University, China. His research interest mainly focuses on computer vision.



**Yadan Han** received the BEng degree from Henan University of Urban Construction, China, in 2021. She is currently pursuing the MEng degree at Guangxi Normal University, China. Her research interests include knowledge representation learning and knowledge graph completion.



**Guoqiu Wen** received the PhD degree from Guangxi Normal University, China, in 2022. She is currently an associate professor at Guangxi Normal University, China. Her research interests include machine learning, feature learning, graph learning, and data mining.



**Taotao Qiu** received the MEng degree in processing and storage of agriculture products from Southwest University, China, in 2009, and the PhD degree in food safety and nutrition from South China Agricultural University, China, in 2018. He is an associate professor at College of Physical Education and Health, Guangxi Normal University, China. His research interest mainly focuses on machine learning.