PMAA: A Progressive Multi-Scale Attention Autoencoder Model for High-Performance Cloud Removal from Multi-Temporal Satellite Imagery

Xuechao Zou[a], **Kai Li**[b;*], **Junliang Xing**[b], **Pin Tao**[a,b;**] and **Yachao Cui**[a]

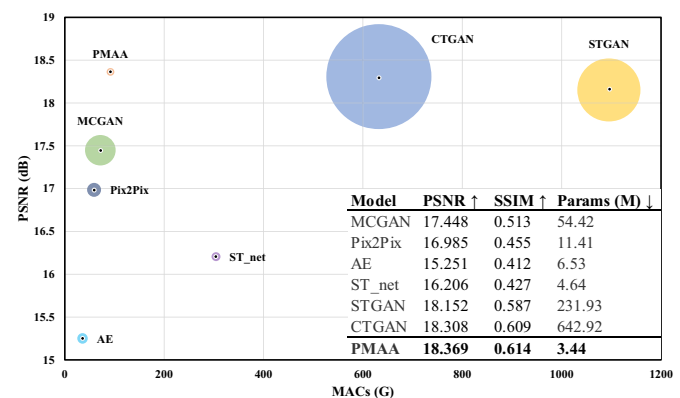[a]Department of Computer Technology and Applications, Qinghai University, Xining, China
[b]Department of Computer Science and Technology, Tsinghua University, Beijing, China

Abstract. Satellite imagery analysis plays a pivotal role in remote sensing; however, information loss due to cloud cover significantly impedes its application. Although existing deep cloud removal models have achieved notable outcomes, they scarcely consider contextual information. This study introduces a high-performance cloud removal architecture, termed Progressive Multi-scale Attention Autoencoder (PMAA), which concurrently harnesses global and local information to construct robust contextual dependencies using a novel Multi-scale Attention Module (MAM) and a novel Local Interaction Module (LIM). PMAA establishes long-range dependencies of multi-scale features using MAM and modulates the reconstruction of fine-grained details utilizing LIM, enabling simultaneous representation of fine- and coarse-grained features at the same level. With the help of diverse and multi-scale features, PMAA consistently outperforms the previous state-of-the-art model CTGAN on two benchmark datasets. Moreover, PMAA boasts considerable efficiency advantages, with only 0.5% and 14.6% of the parameters and computational complexity of CTGAN, respectively. These comprehensive results underscore PMAA's potential as a lightweight cloud removal network suitable for deployment on edge devices to accomplish large-scale cloud removal tasks. *Our source code and pre-trained models are available at https://github.com/XavierJiezou/PMAA.*

1 Introduction

With the rapid development of remote sensing technologies, satellite imagery has been widely applied in various fields, such as grassland monitoring [33], ground target detection [36], land cover classification [1, 34], *etc*. Nonetheless, due to weather conditions, clouds often disrupt the imaging process of optical sensors carried on satellites, resulting in information loss and image quality degradation. Cloud removal from satellite imagery attempts to reconstruct the original information in cloud-covered areas to solve the above problems. It is a critical preprocessing step that significantly affects the effective use of satellite imagery.

Recently, convolutional neural networks (CNNs [16]) and generative adversarial networks (GANs [7]) have shown significant improvements in cloud removal performance for satellite imagery. Among these, mono-temporal cloud removal methods [6, 17] generate a corresponding cloud-free image using a single cloudy image.

---

* Xuechao Zou and Kai Li contributed equally to this work.
** Corresponding Author. Email: taopin@tsinghua.edu.cn.

Figure 1. Performance and efficiency comparison on the *Sen2_MTC_New* dataset. Compared to existing methods, our proposed PMAA achieves the SOTA performance while being computationally efficient and having few parameters. The area of the circle indicates the number of model parameters. Complete results are shown in Table 2.

Model	PSNR ↑	SSIM ↑	Params (M) ↓
MCGAN	17.448	0.513	54.42
Pix2Pix	16.985	0.455	11.41
AE	15.251	0.412	6.53
ST_net	16.206	0.427	4.64
STGAN	18.152	0.587	231.93
CTGAN	18.308	0.609	642.92
PMAA	**18.369**	**0.614**	**3.44**

These methods exhibit stable performance in cloud removal from satellite imagery. However, when cloud coverage is extensive, sufficient information cannot be effectively obtained, making it challenging to generate cloud-free satellite imagery and often impossible.

As remote sensing technology advances, satellite revisitation periods to the same location become increasingly shorter. We can quickly obtain multiple images of the same area captured by satellites at different times. Recently, some researchers have explored utilizing multi-temporal satellite images and successfully applied them to cloud removal to enhance performance. The multi-temporal cloud removal method uses multiple cloudy images at the same location as inputs and generates a cloud-free image using spatial and temporal information. Among these, [32] proposed a CNN-based autoencoder using multi-temporal satellite imagery to remove clouds. [2] combined cloud detection network to improve performance. [31] treated the cloud removal problem as a challenge of conditional image synthesis and proposed a spatiotemporal generative network for cloud removal. [12] proposed a Transformer-based GAN for cloud removal, increasing the accuracy of generated cloud-free images.

However, cloud removal methods suffer from more issues. Firstly, almost all methods do not consider the model complexity, poten-

tially resulting in additional time consumption, especially for large-resolution satellite imagery. Secondly, existing methods [6, 13, 32, 2, 31, 12] do not consider progressively generating cloud-free images but only perform a single stage of reconstruction, which may affect the reconstruction of spatial details in the image [41]. Thirdly, they generally only consider the impact of local information on generating cloud-free images without considering the combination of global and local information. Therefore, the challenge remains to efficiently exploit global and local information for the cloud removal task.

To address the mentioned issues, we present a high-performance Progressive Multi-scale Attention Autoencoder (PMAA) that effectively captures fine- and coarse-grained features across different scales, as depicted in Figure 3. The PMAA mainly comprises two novel components, the Multi-scale Attention Module (MAM) and the Local Interaction Module (LIM). Firstly, the MAM aggregates multi-scale features to acquire global attention through multi-temporal satellite images and then modulate the multi-scale features. Secondly, we use the LIM to connect the local and global features extracted by the MAM, which reconstructs the fine-grained image structure. To further enhance the fidelity of the generated images, we propose a novel progressive learning approach for iterative refinement. The main contributions of our work can be summarized in three key aspects.

- We propose a novel lightweight architecture, named PMAA, for removing clouds from satellite imagery, which mainly incorporates two carefully designed components: MAM and LIM.
- We design an efficient MAM that effectively compensates for the loss of accuracy in reconstructing cloud-free images by aggregating features of different spatial resolutions into global features.
- We design a new reconstruction component, LIM, to enhance the fine-grained generation of cloud-free images through selectively fusing global and local features.

Based on the above main contributions, we obtain a high-performance model for cloud removal from satellite imagery. Our extensive experiments on the PMAA demonstrate its superiority in the task of cloud removal. Specifically, PMAA consistently achieves state-of-the-art (SOTA) performance on the *Sen2_MTC_Old* [31] and *Sen2_MTC_New* [12] datasets compared to previous methods. Furthermore, we also demonstrate the efficiency of PMAA, which achieves the optimal performance-efficiency trade-off. Compared to the previous SOTA model, PMAA saves approximately 99.5% of parameters and 85.4% of the computational cost, as shown in Figure 1.

2 Related work

2.1 Cloud removal

With the development of deep learning, cloud removal methods for satellite imagery have increasingly become the focus of current research. Existing methods can be classified into two types: mono-temporal and multi-temporal. The mono-temporal-based methods [6, 17, 3] will have a much faster inference time because its input uses only a single cloudy image. *Enomoto et al.* [6] proposed a CGAN-based method to achieve thin cloud removal on multispectral remote sensing data. *Lin et al.* published a mono-temporal dataset and used Pix2Pix [13] as the baseline for cloud removal. Pan introduced a spatial attention mechanism in GAN to enhance the information recovery of cloud regions to generate better quality. *Czerkawski et al.* [3] used an internal learning regime based on the deep image before obtaining the ability to inpaint the cloud-affected regions.
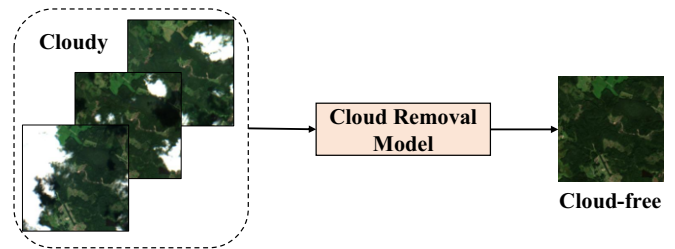
Figure 2. A pipeline for cloud removal task using multi-temporal satellite imagery. Input is multiple cloudy images of the same location and adjacent moments, and output is a cloud-free image of the corresponding location.

However, when many clouds cover the satellite image, the mono-temporal-based methods may not obtain precise results, which limits its application in practical scenarios. Multi-temporal-based methods [32, 2, 31, 12] obtain better results than mono-temporal-based methods using multi-temporal cloudy images to reconstruct a single cloud-free image. *Sintarasirikulchai et al.* [32] proposed cloud removal using convolutional autoencoders by training on a multi-temporal remote sensing dataset. *Chen et al.* [2] combined cloud detection techniques to perform cloud removal by fusing spatiotemporal features of multi-temporal data. *Sarukkai et al.* [31] treated the cloud removal problem as a conditional image synthesis challenge and proposed a spatiotemporal generative network for cloud removal. *Huang et al.* [12] proposed a Transformer-based GAN for de-clouding. Although they recover high-quality cloud-free images, the application's inference process takes more time. In contrast, our proposed PMAA focuses on multi-temporal satellite images' global and local features and efficiently reconstructs cloud-free images. Furthermore, PMAA dramatically reduces the computational cost of the model, making practical applications possible.

2.2 Attention mechanism

Attention mechanism originates from natural language processing (NLP), such as language modeling [14, 21, 15], machine translation [35], and generative tasks [11, 18]. They estimate the relationship between current and global features via the attention mechanism. Recently, the computer vision field has also successfully used attention mechanisms to improve model performance. [4] proposed using a transformer-based model to learn normalized grayscale sketch tensor space for accomplishing painting tasks. This attention-based model can significantly learn global structures with long-range dependencies, which helps to address the limitations of Convolutional Neural Networks (CNN) in recovering the overall structure of images. The UQ-Transformer [20] takes unquantized feature vectors from the encoder as an input and uses the quantized tokens of unmasked patches as prediction targets, thereby reducing information loss and improving prediction accuracy. However, these methods always transfer the attention map to deeper layers, which can lead to shallow features being less affected by attention and bring limited performance improvements. In addition, existing methods [5, 22] tend to divide the image into non-overlapping patches to reduce the computational cost, which is unreliable for cloud removal. This is because the image's cloud distribution is not uniform, leading to inconsistent cloud occupancy in different patches.

2.3 Progressive learning

Progressive learning aims to split complex processes (*e.g.*, direct reconstruction of cloud-free images) into multiple easier and smaller stages (*e.g.*, multi-step reconstruction of cloud-free images) to improve model performance and is widely considered in visual tasks and speech tasks, such as image synthesis [9], image super-resolution [38, 19], and speech separation [11, 18]. In addition, progressive learning is more in line with how humans perceive images because the human visual system does not process the whole scene simultaneously. Instead, it gradually focuses its attention on the part of the interest of the image and ignores the irrelevant details. It can combine information from different regions to reconstruct the complete scene in the brain [26, 28, 29].

3 Method

3.1 Overall pipeline

As shown in Figure 2, the algorithm of multi-temporal cloud removal is expected to generate a cloud-free satellite image from multiple cloudy satellite images (same location, adjacent time). We denote three cloudy satellite images as $\{\mathbf{X}_i \in \mathbb{R}^{4 \times H \times W} | i = 1, 2, 3\}$, where H and W are the image's height and width, respectively, and "4" denotes the four Spectral channels (RGB and infra-red). And we denote a cloud-free image at the current location as $\mathbf{y} \in \mathbb{R}^{4 \times H \times W}$. We assume that for an arbitrary location, $\mathbf{X}_i$ changes slowly over time and that the cloud cover position in the image varies.

Given the input multi-temporal satellite images $\{\mathbf{X}_i | i = 1, 2, 3\}$, we firstly preprocess them using a weight shared bottleneck consisting of several convolutions with residual connection, which serve as input for the cloud removal autoencoder. Secondly, the encoder in the cloud removal autoencoder downsamples through convolutional layers with the stride size of 2×2, producing features with different spatial resolutions. Then they are aggregated to obtain fine- and coarse-grained representations. Thirdly, aggregated multi-scale features are fed into the MAM to obtain global attention and modulate multi-scale features. Then, the LIM in the decoder connects local and global features to reconstruct fine-grained image structures. Finally, we use a novel progressive learning method to cycle PMAA's cloud removal autoencoder to generate approximate cloud-free images.

3.2 High-performance cloud removal autoencoder

We design a novel high-performance cloud removal autoencoder (as shown in Figure 3) that receives the spatiotemporal features $\mathbf{U}_c \in \mathbb{R}^{12 \times H \times W}$ as input, which are obtained by concatenating $\{\mathbf{U}_i | i = 1, 2, 3\}$ on the channel dimension. After progressive refinement, the cloud removal module ultimately yields a cloud-free image $\bar{\mathbf{y}}$ at the current location. The cloud removal module is described in the following sections.

3.2.1 Encoder

To consider data dimensionality reduction and mimicking human cognitive processes, we commonly deepen image features through downsampling while preserving as much relevant information as possible. To this end, we introduce an encoder (Figure 3(a)) capable of extracting multi-scale features. To achieve this, we employ multiple depth-wise separable convolutional layers [10] with a kernel size of 3×3 and stride of 2×2 for reducing spatial image scale and increasing receptive field. After downsampling N times, we obtain

$N + 1$ multi-scale features $\mathbf{F}_i \in \mathbb{R}^{C \times \frac{H}{2^i} \times \frac{W}{2^i}} | i = 0, ..., N\}$ with the incremental receptive field, where C denotes the features channels. All convolutional layers are followed by instance normalization and ReLU activation function [27]. Finally, the multi-scale features $\mathbf{F}_i$ are fed into the following Multi-scale Attention Module.

3.2.2 Multi-scale attention module

To endow the model with the capability of perceiving both fine- and coarse-grained visual features, we introduce a multi-scale attention module (MAM, Figure 3(b)) composed of three components: 1) Multi-scale Fusion; 2) Transformer Layer; 3) Selective Attention. The MAM works as follows.

Multi-scale fusion. First, we craft a multi-scale fusion approach without any parameters and with insignificant additional computation costs. It compresses all features $\{\mathbf{F}_i | i = 0, ..., N\}$ with the scale $\left(\frac{H}{2^i} \times \frac{W}{2^i}\right)$ to the uniform scale $\left(\frac{H}{2^N}, \frac{W}{2^N}\right)$ using the adaptive average pooling layers and then fuse them by a summation operation to obtain a multi-scale representation

$$\mathbf{F}_{\text{ms}} = \sum_{i=0}^{N} H(\mathbf{F}_i), \tag{1}$$

where $H(\cdot)$ represents adaptive average pooling layer. Finally, we use the $\mathbf{F}_{\text{ms}}$ as the input to the transformer layer in the next stage.

Transformer layer. Limited by the receptive fields of CNNs, some methods [13, 6, 31] struggle to acquire enough contextual information for cloud removal, particularly when cloud cover is extensive. This is because an individual pixel and its receptive field may all fall in the cloudy region, preventing it from noticing information outside the cloud area and severely hindering the recovery of the cloud-free image. Therefore, we introduce self-attention [35] to encode spatial information to establish long-range dependency. So far, there are many vision transformers based on self-attention, such as ViT [5] and Swin Transformer [22]. To balance performance and efficiency, we adopt a simple self-attention implemented by using a convolutional modulation operation that uses large kernel convolution to avoid the problem of time-consuming and complex computation of the attention matrix (see Figure 4)

$$\mathbf{F}_a = \mathbf{F}_{\text{ms}} + \alpha \mathbf{W}_3 (\text{DConv}_{k \times k}(\mathbf{W}_1 \mathbf{F}_{\text{ms}}) \odot \mathbf{W}_2 \mathbf{F}_{\text{ms}}), \tag{2}$$

where $\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3$ are linear layers and α is a learnable parameter, and $\text{DConv}_{k \times k}$ denotes depth-wise convolutional layer with a large kernel size. Then, a residual connection [8] is added after self-attention to reduce information loss. The self-attention is immediately followed by a feed-forward network (FFN), which consists of one depth-separable convolution layer and two linear layers.

$$\mathbf{F}_g = \mathbf{F}_a + \beta \mathbf{V}_2 (\mathbf{V}_1 \mathbf{F}_a + \text{DConv}_{k \times k}(\mathbf{V}_1 \mathbf{F}_a)), \tag{3}$$

where $\mathbf{V}_1, \mathbf{V}_2$ are linear layers and β is a learnable parameter. In summary, we obtain a feature $\mathbf{F}_g$ with global information by processing a transformer layer.

Selective attention. We use $\mathbf{F}_g$ as global attention to perform adaptive feature recalibration on $\mathbf{F}_i$ before integrating it into the decoder. This is because the earlier neural network layers possess rich low-level texture features, while the deeper layers have high-level semantic information. Specifically, we upsample $\mathbf{F}_g$ through nearest-neighbor interpolation to obtain the same spatial dimension
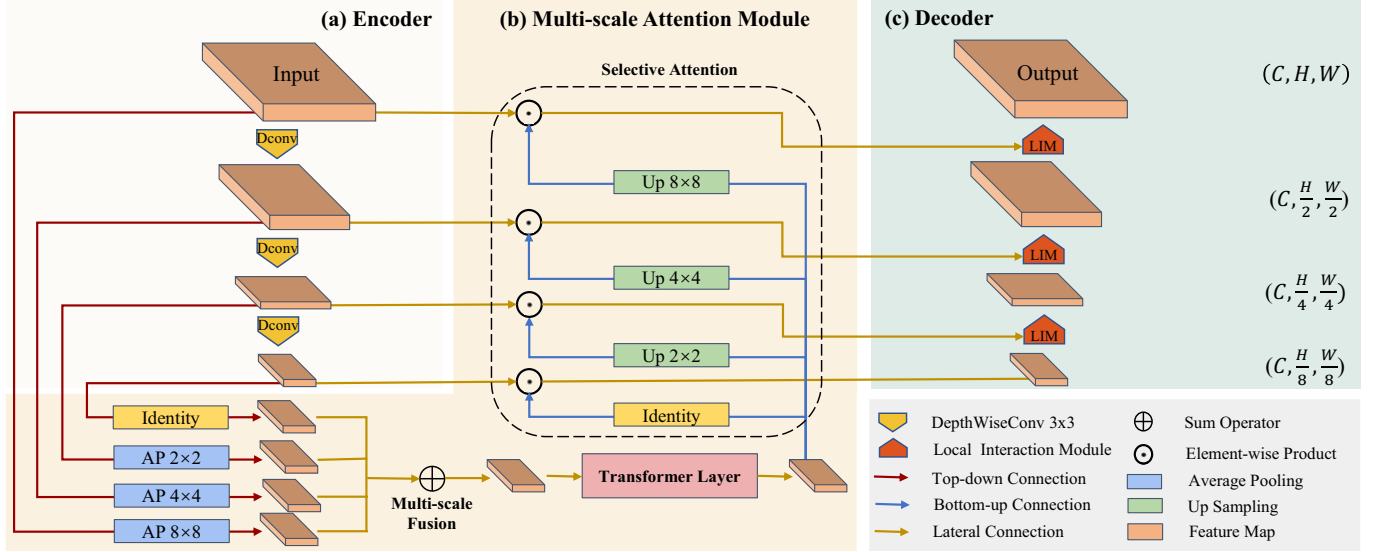
Figure 3. Overview of our proposed high-performance cloud removal autoencoder. In the encoder, we downsample the input image N times. Then, the multi-scale features are fused by averaging pooling and summation operations. A simplified transformer layer processes the fused features to obtain global attention, which is used to modulate the multi-scale features. In the reconstruction process (decoder), we use the local interaction module to recover more details.

as $\mathbf{F}_i$. Then, we obtain the modulated feature $\{\mathbf{F}'_i \in \mathbb{R}^{c \times \frac{H}{2^i} \times \frac{W}{2^i}} | i = 0, ..., N\}$ through affine transformation

$$\mathbf{F}'_i = \phi(\sigma(\mathbf{Z}_1(\mathbf{F}_g))) \odot \mathbf{Z}_2(\mathbf{F}_i) + \phi(\mathbf{Z}_3(\mathbf{F}_g)), \quad (4)$$

where $\mathbf{Z}_1, \mathbf{Z}_2, \mathbf{Z}_3$ are linear layers, σ represents the sigmoid activation function, and ϕ represents the nearest-neighbor interpolation.

3.2.3 Decoder

The global feature $\mathbf{O}_i$ has a larger receptive field and contains high-level information. In contrast, the local feature $\mathbf{F}'_{i+1}$ contains low-level texture information, but the receptive field is limited. We design a local interaction module (LIM) to fuse these two features effectively. The local interaction module (see Figure 4(b)) is the core component in the decoder (see Figure 3(c)), which gradually restores image resolution through the previously modulated features $\mathbf{F}'_i$

Specifically, we use $\mathbf{O}_i$ after upsampling as the weights of $\mathbf{F}'_{i+1}$ to obtain more robust local features. At the same time, $\mathbf{O}_i$ are convolutionally modulated and then residual concatenated with $\mathbf{F}'_{i+1}$ to obtain a refined feature representation $\mathbf{O}_{i+1}$ containing both global and local information:

$$\mathbf{O}_{i+1} = \phi(\sigma(\mathbf{D}_1(\mathbf{O}_i))) \odot \mathbf{D}_2(\mathbf{F}'_{i+1}) + \phi(\mathbf{D}_3(\mathbf{O}_i)), \quad (5)$$

where $\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3$ are three depth-wise convolutional layers, σ represents the sigmoid activation function, and ϕ represents the nearest-neighbor interpolation, some details such as the normalization layer are omitted. Finally, we shall take the feature map with the highest resolution as the output of the current cloud removal model.

3.3 Progressive learning

Progressive learning (PL) has been introduced to image inpainting [25, 37] and image restoration [41, 40] tasks and has achieved superior performance. Multi-temporal cloud removal is a complex task between image inpainting and restoration. Instead of attempting

a direct transformation from multi-temporal cloudy images to a single cloud-free image, we adopt a strategy of breaking down the cloud removal process into several smaller, more tractable steps. These preliminary stages assist in laying the foundation for the latter stages of the training process:

$$\mathbf{Q}_{t+1} = S(\mathbf{Q}_t \oplus \varphi(\mathbf{U}_c)), \quad (6)$$

where $\mathbf{Q}_t$ denotes the output of the cloud removal module at time t, S represents a cloud removal module, and φ represents a convolutional layer with kernel size of 1×1 to do the work of channel transformation. Although too many stages may cause the network to be too deep to converge, increasing the model's parameters and computational cost. We explored the effects of different stages in the ablation experiments (see Figure 5). We chose the optimal number of stages to balance performance and efficiency.

3.4 Loss function

The cloud removal network is optimized to improve the model's ability to eliminate clouds and achieve high-fidelity cloud-free image generation. To align the divergence between the generated cloud-free image and the ground truth, we calculate the L1 loss between them

$$\mathcal{L}_{L1}(F) = ||\mathbf{y} - F(x)||_1, \quad (7)$$

where x represents multi-temporal cloudy images, F represents cloud removal model, y represents the ground truth, and $F(x)$ denotes the estimated cloud-free image.
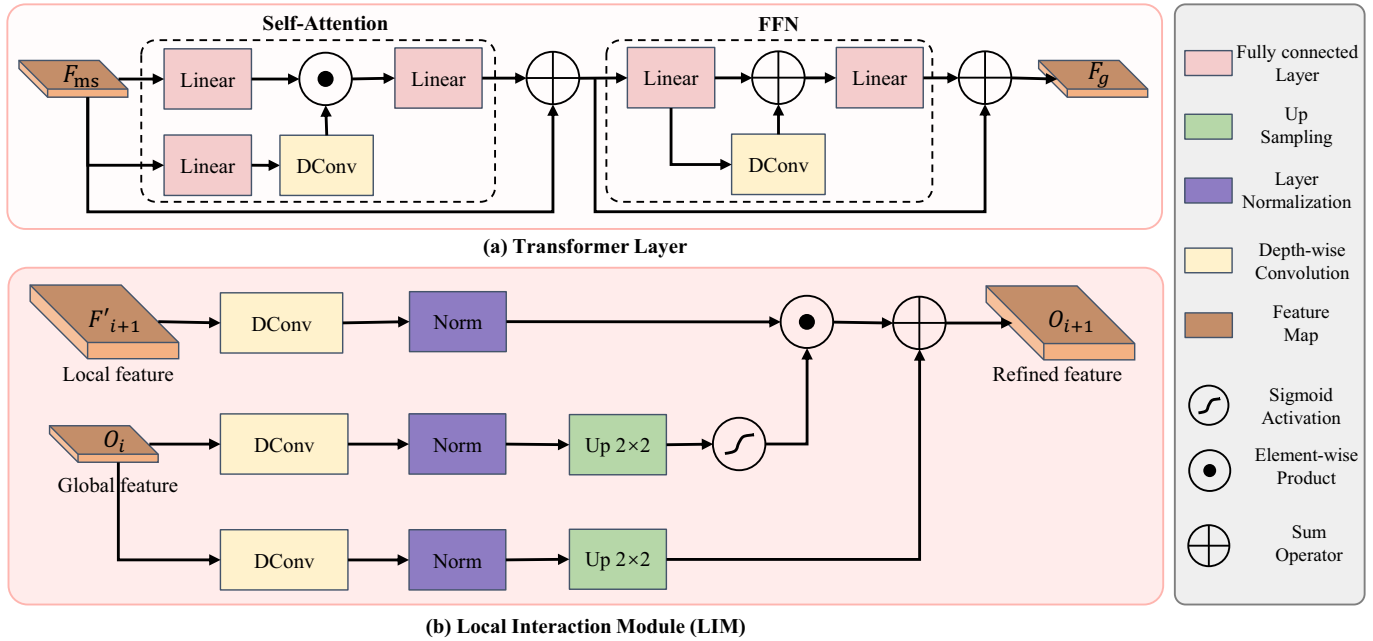
4 Experiments

4.1 Datasets

To validate the efficacy of PMAA and its novel components, we conduct experiments on two widely-recognized cloud removal datasets.

Sen2_MTC_Old. This dataset [31] is created from publicly-available Sentinel-2 images. It contains 945 different tiles with a total

Figure 4. Structure diagram of the transformer layer and local interaction module. (a) Transformer Layer: we adopt a simple self-attention implemented by using a large kernel convolution (the kernel size of "DConv" in self-attention is set to 11×11) to obtain long-range contextual information; the self-attention is immediately followed by a feed-forward network (FFN). Some details are omitted for simplicity, *e.g.*, reshaping the feature maps. (b) Local Interaction Module (LIM): we use convolutional modulation to fuse local features from skip connections and global features from the transformer layer to obtain a refined feature.

of 3130 image pairs. Every three cloudy images correspond to one cloud-free image, where each image has size $(w, h) = (256, 256)$, the number of channels $C = 4$ (RGB and infra-red), and pixel values in the range $[0, 255]$. The pixel values of each image are normalized to $[0, 1]$ before input and then transformed to $[-1, 1]$ by a mean and variance of 0.5. This dataset divides into a training set, a validation set, and a test set in the ratio of $8 : 1 : 1$.

Sen2_MTC_New. This dataset [12] is also created from the Sentinel-2 images, which contains about 50 non-overlapping tiles with about 70 pairs of images per tile. Its settings are consistent with the *Sen2_MTC_Old* dataset, except that the pixel value range is $[0, 10000]$. Compared to the *Sen2_MTC_Old* dataset, it has higher resolution and annotation quality. This dataset divides into training, validation, and test sets in the ratio of $7 : 1 : 2$.

4.2 Implementation details

Training settings. Initially, we normalize all images to [-1, 1], then concatenate multiple cloudy images along the channel dimension, feeding them into several bottleneck layers consisting of convolutions to extract features. The downsampling and upsampling counts of the encoder and decoder are set to 4. The channel count of the hidden layer is set to 32. Finally, PMAA predicts cloud-free image through a convolution layer with a kernel size of 3×3 and a stride of 1×1. During the training phase, we use AdamW [24] optimizer with an initial learning rate of 5×10^{-4} and weight decay 1×10^{-5}, and a cosine decay [23] learning rate schedule. We train all models for 100 epochs, with a batch size of 4, and save the model with the best SSIM value on the validation set for testing on the test set. We conduct experiments on a machine with $4 \times$ NVIDIA GeForce RTX 3090 (24GB memory).

Evaluation metrics. In all experiments, we report the Peak

Signal-to-Noise Ratio (PSNR, dB) and Structual Similarity Index Measure (SSIM [39]) of the test set to evaluate the precision of the generated cloud-free images. To evaluate model efficiency, we report the number of parameters (M) and multiply–accumulate operations (MACs, G) for all models. The result is calculated via the ptflops[1].

4.3 Ablation studies

To quantitatively assess the individual impact of each component, we conduct a series of extensive ablation experiments on the *Sen2_MTC_New* dataset, which possesses superior annotation quality compared to the *Sen2_MTC_Old* dataset.

4.3.1 Multi-scale fusion strategy

We investigate the consequences of various multi-scale feature fusion approaches on cloud removal performance, as depicted in Table 1. In contrast to abstaining from feature fusion (utilizing the feature map with the most reduced resolution as the transformer layer input), the summation operation emerges as the optimal feature fusion strategy, culminating in a PSNR enhancement of 0.168 and SSIM enhancement of 0.013. It is worth noting that unsuitable feature fusion strategies (such as channel-dimension concatenation) may yield not only inferior results but also elevate computational complexity.

4.3.2 Self-attention implementation

As outlined in Table 1, we examine the impact of various self-attention strategies on cloud removal performance. Compared to the transformer layer based on patch (like Swin-Transformer [22]), the
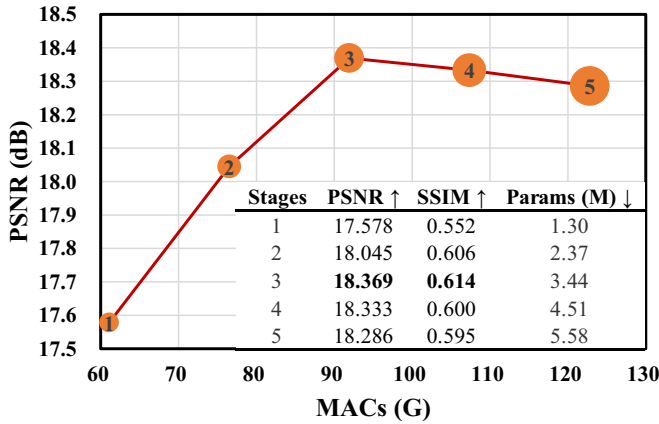
---

[1] https://github.com/sovrasov/flops-counter.pytorch

Table 1. Ablation studies of MAM and LIM on the *Sen2_MTC_New* dataset. "✓" indicates that this part is used, while "×" denotes the absence of its use.

MAM					LIM	PSNR ↑	SSIM ↑	Params (M) ↓	MACs (G) ↓
Multi-scale fusion		Transformer layer		Selective attention					
Concat	Sum	Patch	Nonpatch						
×	×	×	×	×	×	17.441	0.581	2.71	90.30
×	×	×	✓	✓	✓	18.201	0.601	3.44	91.91
✓	×	×	✓	✓	✓	18.045	0.606	3.73	92.01
×	✓	×	×	✓	✓	17.809	0.589	2.83	91.78
×	✓	✓	×	✓	✓	18.032	0.599	3.25	91.85
×	✓	×	✓	×	✓	18.300	0.608	3.43	91.87
×	✓	×	✓	✓	×	17.590	0.597	3.32	90.56
×	✓	×	✓	✓	✓	**18.369**	**0.614**	3.44	91.94

Table 2. Quantitative comparison of cloud removal performance and efficiency between PMAA and existing models on *Sen2_MTC_Old* and *Sen2_MTC_New*.

Methods	Sen2_MTC_Old		Sen2_MTC_New		Params (M) ↓	MACs (G) ↓
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑		
MCGAN [6]	21.146	0.481	17.448	0.513	54.42	71.56
Pix2Pix [13]	22.894	0.437	16.985	0.455	11.41	58.94
AE [32]	23.957	0.800	15.251	0.412	6.53	**35.72**
ST_net [2]	26.321	0.834	16.206	0.427	4.64	304.31
STGAN [31]	26.186	0.734	18.152	0.587	231.93	1094.94
CTGAN [12]	26.264	0.808	18.308	0.609	642.92	632.05
PMAA (Ours)	**27.377**	**0.861**	**18.369**	**0.614**	**3.44**	91.94

Stages	PSNR ↑	SSIM ↑	Params (M) ↓
1	17.578	0.552	1.30
2	18.045	0.606	2.37
3	**18.369**	**0.614**	3.44
4	18.333	0.600	4.51
5	18.286	0.595	5.58

Figure 5. Quantitatively compare the impact of the number of progressive learning stages. The area of the circle indicates the number of model parameters, whereas larger circles indicate more parameters. The numbers in circles indicate the number of different stages.

nonpatch-based transformer layer (PMAA's transformer layer) obtains better performance, with PSNR and SSIM rising by 0.337 and 0.015, respectively. One possible explanation for this substantial advancement could be that the patch operation may lead to some patches lacking cloud occlusion while others being entirely cloud occlusion, thus rendering PMAA incapable of learning efficiently.

4.3.3 Selective attention

We also investigate the impact of the selective attention module in MAM on cloud removal performance, as shown in Table 1. Note that when the selective attention module is not included in the model, the skip connections are consistent with those of U-Net [30]. The results demonstrate that utilizing the selective attention module leads to improved cloud removal performance (0.069 increase in PSNR and 0.006 increase in SSIM), while introducing an extremely low parameter and computational complexity (approximately 0.01M and

0.07 GMACs). This can be explained by the fact that shallow feature extraction in the network contains some redundant information, whereas the adoption of selective attention mechanisms can effectively filter out such useless information while enhancing the useful information, resulting in a more refined feature representation.
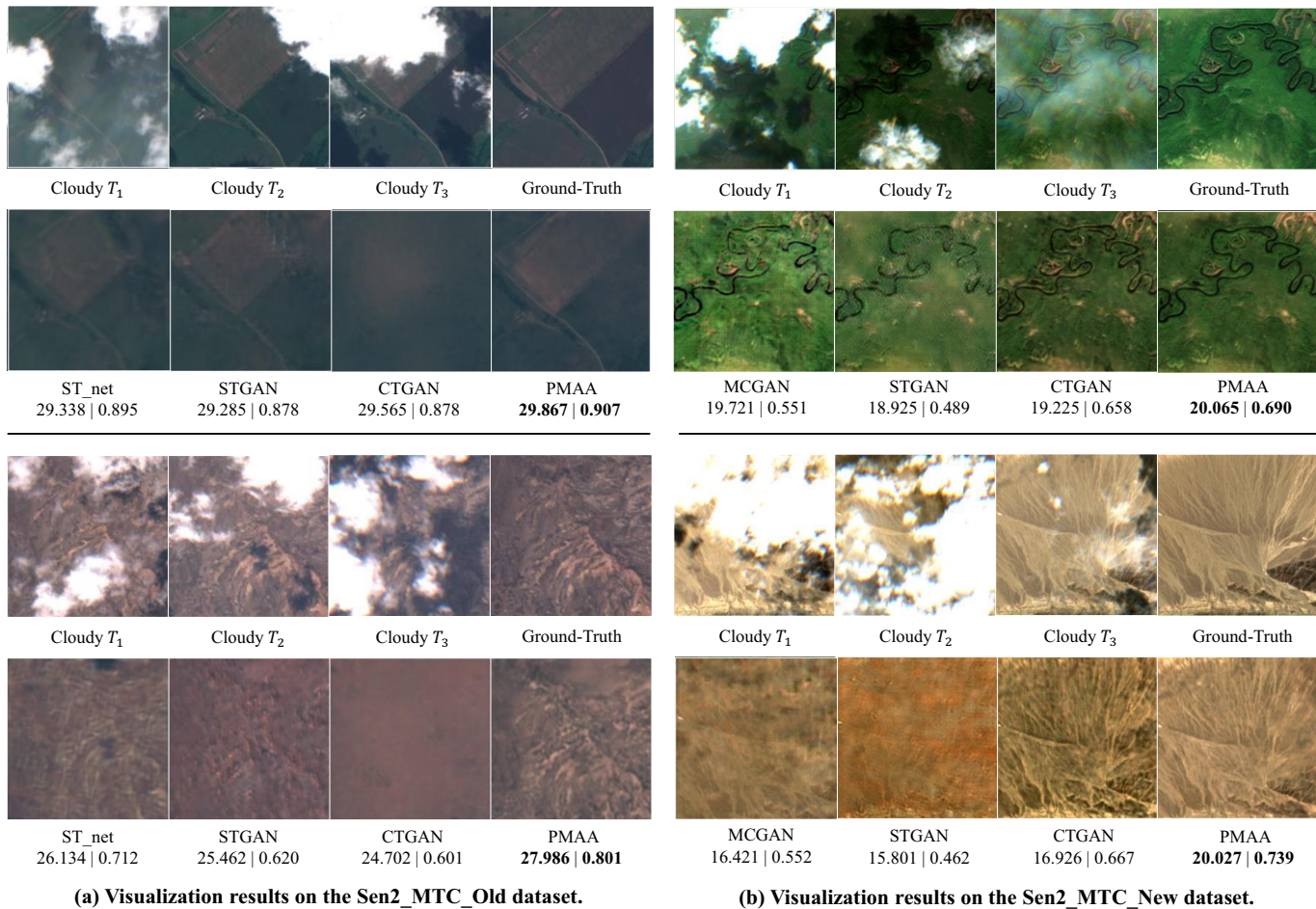
4.3.4 Local interaction module

Furthermore, we examine the ramifications of incorporating the LIM within the decoder on cloud removal efficacy. As delineated in Table 1, leveraging the Local Interaction Module yields superior performance (0.779 and 0.017 increase in PSNR and SSIM, respectively) in comparison to its absence (as observed in the initial upsampling operation in U-Net [30]). This underscores the significance of local feature amalgamation in directing attention toward pertinent areas.

4.3.5 Progressive learning

Expanding the quantity of cloud removal autoencoders augments the progressive learning procedure, consequently boosting the model's expressive capabilities. Nonetheless, this simultaneously incurs additional computational expenses. In Figure 5, we scrutinize the influence of the stage count in PMAA on model efficacy and performance. A satisfactory equilibrium between performance and efficiency is attained when the stage count within PMAA is assigned a value of 3. If the stage count perpetually increases, the model exhibits excessive intricacy and fails to converge, while the cross-stage transfer may precipitate information attrition, culminating in suboptimal results. Consequently, we adopt a default configuration comprising three stages in the conducted experiments.

4.4 Comparison with the state-of-the-arts

We conduct extensive experiments to compare the performance and efficiency of our proposed PMAA with existing models on *Sen2_MTC_Old* and *Sen2_MTC_New* datasets. Our model achieves

Cloudy T_1	Cloudy T_2	Cloudy T_3	Ground-Truth

ST_net	STGAN	CTGAN	PMAA
29.338 | 0.895	29.285 | 0.878	29.565 | 0.878	**29.867 | 0.907**

Cloudy T_1	Cloudy T_2	Cloudy T_3	Ground-Truth

ST_net	STGAN	CTGAN	PMAA
26.134 | 0.712	25.462 | 0.620	24.702 | 0.601	**27.986 | 0.801**

(a) Visualization results on the Sen2_MTC_Old dataset.

Cloudy T_1	Cloudy T_2	Cloudy T_3	Ground-Truth

MCGAN	STGAN	CTGAN	PMAA
19.721 | 0.551	18.925 | 0.489	19.225 | 0.658	**20.065 | 0.690**

Cloudy T_1	Cloudy T_2	Cloudy T_3	Ground-Truth

MCGAN	STGAN	CTGAN	PMAA
16.421 | 0.552	15.801 | 0.462	16.926 | 0.667	**20.027 | 0.739**

(b) Visualization results on the Sen2_MTC_New dataset.

Figure 6. Visualization results on *Sen2_MTC_Old* and *Sen2_MTC_New*. The left and right sides of "|" indicate the values of PSNR and SSIM, respectively.

SOTA cloud removal performance using a very small number of parameters, which can be called a lightweight network.

Performance of cloud removal. we present a comprehensive comparison of the quantitative results of our proposed method with existing state-of-the-art methods on the *Sen2_MTC_Old* and *Sen2_MTC_New* datasets, as detailed in Table 2. On both datasets, PMAA achieves consistent SOTA performance on PSNR and SSIM, demonstrating our method's superiority. On two benchmark datasets, PMAA performs much better than the previous SOTA model CT-GAN and other methods with only 0.5% of the number of parameters compared to CTGAN.

The visualization results are shown in Figure 6, which presents a qualitative comparison of our method against existing approaches on representative examples from two datasets. We observe that PMAA can consistently generate more detailed structures and demonstrates improved robustness in cloud removal on two datasets.

Efficiency of cloud removal. We compare with existing methods regarding the number of model parameters and MACs (reflecting the computational complexity). In Table 2, we observe that PMAA has a superior efficiency with 0.5% and 14.6% of the previous SOTA model CTGAN's number of parameters and MACs, respectively. Our efficiency gain is attributed to two components: MAM and LIM. First, the encoder uses depth-wise convolution layers for downsampling, and MAM fuses multi-scale features using parameter-free pooling layers. This is more efficient than CTGAN, which distributes

the computational cost in each layer. Second, our transformer layer in MAM does not compute the similarity matrix and uses only the large convolutional kernel to extract features since it calculates at the minimum resolution scale, which is sufficient to extract global features. Third, LIM in the decoder uses very few parameters to fuse global and local features to obtain refined feature representations. The extremely low parameter number and computational complexity suggest PMAA's excellent deployability to low-resource devices.

5   Conclusion

To address the issue of low efficiency in current cloud removal methods, this paper presents a high-performance network refer as PMAA for cloud removal. PMAA leverages the autoencoder architecture with an efficient multi-scale attention module to capture global contextual information and guide the local interaction module in the decoder to reconstruct cloud-free images. Meanwhile, PMAA uses progressive learning to improve the model performance further. Sufficient experiments demonstrate that PMAA has essential implications for grounding the cloud removal model in practical deployments and is expected to achieve superior performance in large-scale and batch-processing tasks. In addition, owing to the efficacy and lightweight, we believe that the modules we have introduced can be seamlessly transplanted to other related problems, such as image restoration (e.g., rain, snow, fog, and noise removal) and other generative tasks.

Acknowledgements

References

[1] Manuel Carranza-García, Jorge García-Gutiérrez, and José C Riquelme, 'A framework for evaluating land use and land cover classification using convolutional neural networks', *REMOTE SENS-BASEL*, **11**(3), 274, (2019).

[2] Yang Chen, Qihao Weng, Luliang Tang, Xia Zhang, Muhammad Bilal, and Qingquan Li, 'Thick clouds removing from multitemporal landsat images using spatiotemporal neural networks', *TGRS*, **60**, 1–14, (2020).

[3] Mikolaj Czerkawski, Priti Upadhyay, Christopher Davison, Astrid Werkmeister, Javier Cardona, Robert Atkinson, Craig Michie, Ivan Andonovic, Malcolm Macdonald, and Christos Tachtatzis, 'Deep internal learning for inpainting of cloud-affected regions in satellite imagery', *REMOTE SENS-BASEL*, **14**(6), 1342, (2022).

[4] Qiaole Dong, Chenjie Cao, and Yanwei Fu, 'Incremental transformer structure enhanced image inpainting with masking positional encoding', in *CVPR*, pp. 11358–11368, (2022).

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., 'An image is worth 16x16 words: Transformers for image recognition at scale', in *ICLR*, (2020).

[6] Kenji Enomoto, Ken Sakurada, Weimin Wang, Hiroshi Fukui, Masashi Matsuoka, Ryosuke Nakamura, and Nobuo Kawaguchi, 'Filmy cloud removal on satellite imagery with multispectral conditional generative adversarial nets', in *CVPRW*, pp. 48–56, (2017).

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, 'Generative adversarial networks', *COMMUN ACM*, **63**(11), 139–144, (2020).

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, 'Deep residual learning for image recognition', in *CVPR*, pp. 770–778, (2016).

[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel, 'Denoising diffusion probabilistic models', *NeurIPS*, **33**, 6840–6851, (2020).

[10] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, 'Mobilenets: Efficient convolutional neural networks for mobile vision applications', in *CVPR*, (2017).

[11] Xiaolin Hu, Kai Li, Weiyi Zhang, Yi Luo, Jean-Marie Lemercier, and Timo Gerkmann, 'Speech separation using an asynchronous fully recurrent convolutional neural network', in *NeurIPS*, pp. 22509–22522, (2021).

[12] Gi-Luen Huang and Pei-Yuan Wu, 'Ctgan: Cloud transformer generative adversarial network', in *ICIP*, pp. 511–515, (2022).

[13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, 'Image-to-image translation with conditional adversarial networks', in *CVPR*, pp. 1125–1134, (2017).

[14] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova, 'Bert: Pre-training of deep bidirectional transformers for language understanding', in *NAACL-HLT*, pp. 4171–4186, (2019).

[15] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut, 'Albert: A lite bert for self-supervised learning of language representations', in *ICLR*, (2020).

[16] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, 'Deep learning', *Nature*, **521**(7553), 436–444, (2015).

[17] Kyu-Yul Lee and Jae-Young Sim, 'Cloud removal of satellite images using convolutional neural network with reliable cloudy image synthesis model', in *ICIP*, pp. 3581–3585, (2019).

[18] Kai Li, Runxuan Yang, and Xiaolin Hu, 'An efficient encoder-decoder architecture with top-down attention for speech separation', in *ICLR*, (2023).

[19] Kai Li, Shenghao Yang, Runting Dong, Xiaoying Wang, and Jianqiang Huang, 'Survey of single image super-resolution reconstruction', *IET IMAGE PROCESS*, **14**(11), 2273–2290, (2020).

[20] Qiankun Liu, Zhentao Tan, Dongdong Chen, Qi Chu, Xiyang Dai, Yinpeng Chen, Mengchen Liu, Lu Yuan, and Nenghai Yu, 'Reduce information loss in transformers for pluralistic image inpainting', in *CVPR*, pp. 11347–11357, (2022).

[21] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov, 'Roberta: A robustly optimized bert pretraining approach', in *ICLR*, (2020).

[22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo, 'Swin transformer: Hierarchical vision transformer using shifted windows', in *ICCV*, pp. 10012–10022, (2021).

[23] Ilya Loshchilov and Frank Hutter, 'Sgdr: Stochastic gradient descent with warm restarts', in *ICLR*, (2017).

[24] Ilya Loshchilov and Frank Hutter, 'Decoupled weight decay regularization', in *ICLR*, (2018).

[25] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool, 'Repaint: Inpainting using denoising diffusion probabilistic models', in *CVPR*, pp. 11461–11471, (2022).

[26] Yoichi Miyawaki, Hajime Uchida, Okito Yamashita, Masa-aki Sato, Yusuke Morito, Hiroki C Tanabe, Norihiro Sadato, and Yukiyasu Kamitani, 'Visual image reconstruction from human brain activity using a combination of multiscale local image decoders', *Neuron*, **60**(5), 915–929, (2008).

[27] Vinod Nair and Geoffrey E Hinton, 'Rectified linear units improve restricted boltzmann machines', in *ICML*, (2010).

[28] Thomas Naselaris, Ryan J Prenger, Kendrick N Kay, Michael Oliver, and Jack L Gallant, 'Bayesian reconstruction of natural images from human brain activity', *Neuron*, **63**(6), 902–915, (2009).

[29] Shinji Nishimoto, An T Vu, Thomas Naselaris, Yuval Benjamini, Bin Yu, and Jack L Gallant, 'Reconstructing visual experiences from brain activity evoked by natural movies', *CURR BIOL*, **21**(19), 1641–1646, (2011).

[30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, 'U-net: Convolutional networks for biomedical image segmentation', in *MICCAI*, pp. 234–241, (2015).

[31] Vishnu Sarukkai, Anirudh Jain, Burak Uzkent, and Stefano Ermon, 'Cloud removal from satellite images using spatiotemporal generator networks', in *WACV*, pp. 1796–1805, (2020).

[32] Wassana Sintarasirikulchai, Teerasit Kasetkasem, Tsuyoshi Isshiki, Thitiporn Chanwimaluang, and Preesan Rakwatin, 'A multi-temporal convolutional autoencoder neural network for cloud removal in remote sensing images', in *ECTI-CON*, pp. 360–363, (2018).

[33] Felix Stumpf, Manuel K Schneider, Armin Keller, Andreas Mayr, Tobias Rentschler, Reto G Meuli, Michael Schaepman, and Frank Liebisch, 'Spatial monitoring of grassland management using multi-temporal satellite imagery', *ECOL INDIC*, **113**, 106201, (2020).

[34] Xin-Yi Tong, Gui-Song Xia, Qikai Lu, Huanfeng Shen, Shengyang Li, Shucheng You, and Liangpei Zhang, 'Land-cover classification with high-resolution remote sensing images using transferable deep models', *REMOTE SENS ENVIRON*, **237**, 111322, (2020).

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, 'Attention is all you need', *NeurIPS*, **30**, (2017).

[36] Peijin Wang, Xian Sun, Wenhui Diao, and Kun Fu, 'Fmssd: Feature-merged single-shot detection for multiscale objects in large-scale remote sensing imagery', *TGRS*, **58**(5), 3377–3390, (2019).

[37] Tengfei Wang, Hao Ouyang, and Qifeng Chen, 'Image inpainting with external-internal learning and monochromic bottleneck', in *CVPR*, pp. 5120–5129, (2021).

[38] Yifan Wang, Federico Perazzi, Brian McWilliams, Alexander Sorkine-Hornung, Olga Sorkine-Hornung, and Christopher Schroers, 'A fully progressive approach to single-image super-resolution', in *CVPRW*, pp. 864–873, (2018).

[39] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, 'Image quality assessment: from error visibility to structural similarity', *TIP*, **13**(4), 600–612, (2004).

[40] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang, 'Restormer: Efficient transformer for high-resolution image restoration', in *CVPR*, pp. 5728–5739, (2022).

[41] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao, 'Multi-stage progressive image restoration', in *CVPR*, pp. 14821–14831, (2021).