MAE-CG: A MULTI-ATTENTION ENHANCED THIN CLOUD-REMOVAL GENERATIVE ADVERSARIAL NETWORK FOR AIRBORNE IMAGERY

Jayakrishnan A[1]*, Venkatesan M*[1]*, Prabhavathy P*[2]*, Santhanakrishnan S*[3]*, Joshua W*[3]*,Sachin R* [3]

[1] NIT Puducherry, Department of CSE, Karaikal, Puducherry, India
[2]VIT University, Department of IT, Vellore, Tamil Nadu, India
[3]University College of Engineering Arni, Department of CSE, Arni, Tamil Nadu, India

ABSTRACT

Earth observation heavily depend on the spatial-temporal data collected from satellites. However, optical observations can be affected by random thin clouds, which are opaque or semi-transparent in the optical spectrum. This interference impacts the usefulness of satellite-based remote sensing in areas such as resource surveys, vegetation management, and environmental monitoring. To address these challenges, deep learning techniques are evolving to generate cloud-free data from compromised optical observations, despite the complexity of single image cloud-free reconstruction. This paper introduces a cloud-free reconstruction architecture based on a Generative Adversarial Network (GAN) that leverages spatial-attention mechanisms. The proposed Multi Attention Enhanced Thin Cloud-Removal Generative Adversarial Network (MAE-CG) integrates the benefits of the Convolutional Block Attention Module (CBAM) and the Coordinate Attention Module (CAM) to extract superior spatial context. When tested against the RICE-1 dataset, the MAE-CG model demonstrated superior performance in terms of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) compared to existing methods. The model effectively reconstructed cloud-free images by focusing on critical features and spatial details, showing resilience to thin clouds.

Index Terms— Cloud Removal, GAN, CBAM, CAM

1. INTRODUCTION

Remote sensing (RS) images are essential for activities such as land and sea monitoring, disaster detection and management, and agricultural oversight [1]. However, clouds often cover these satellite images, making them difficult to use [2]. Even though clouds can be a problem, thin clouds still allow some ground data to be captured. Because of this, a method to remove thin clouds from images could make the data much more useful [3] [4]. Sentinel-2 is a pair of satellites (Sentinel-2A and Sentinel-2B) that have a high-resolution camera with 13 different types of images, called spectral bands. These bands have different levels of detail: four bands have a 10-meter resolution, six have a 20-meter resolution, and three have a 60-meter resolution. Random clouds make it hard for Sentinel-2 to observe the Earth effectively because of how it captures images [5]. A technique to remove thin clouds from these images could provide much more useful information for various applications [6].

A range of techniques has been developed for cloud removal, such as the multi-spectral image method, multiple image superposition method, image fusion method, histogram matching method, homomorphic filtering method, geometry interpolation method, and various statistical and machine learning methods [7]. These primary methods use past data and process it through statistical and physical correlations to produce a clear, cloud-free image. However, these methods usually remove low-frequency data and adversely affect high-frequency data [3]. Recently, deep-learning approaches, including multiscale interactive fusion networks, multistage self-guided separation networks, and spatial-logical aggregation networks, have gained popularity for cloud removal.

[1] Outlined a cloud detection method based on an iterative refinement strategy. Zhan et al.'s research introduced a CNN network that effectively differentiated between clouds and snow in remote sensing images [4]. However, these methods were not adapted for high-resolution data. Their deep learning approach to cloud detection was effective in detecting [4] and removing clouds from multiple sources in both medium and high-resolution remote sensing images. Recently, CNNs have demonstrated significant abilities in nonlinear function mapping, which has led to their broad application in addressing atmospheric correction challenges [8]. Filtering techniques are widely used for correcting thin clouds, with a notable study applying domain transformation and image filtering [9]. Additionally, an iterative haze optimization algorithm (IHOT) refined the cloud trajectory to generate a fully cloud-free image [10]. A Cloud Aware and Feature Extraction (CAFE) Module was designed by integrating a physical model that accounts for cloud distortion, including factors such as atmospheric light, cloud reflectance, and transmission [11]. Methods such as Laplacian filtering

in the frequency domain and advanced homomorphic filtering are effective for eliminating thin clouds [12]. Exploring frequency domain dependencies in both conventional CNNs and Deep-CNNs has become popular for generating accurate, cloud-free data [13]. Enhancements in CNN architectures, like the multi-scale residual network, have significantly improved the ability to remove thin clouds from remote sensing images [14]. This network was enhanced by creating a multi-scale deep residual network (MDRN) to learn the residuals and directly map between cloudy and cloud-free images [15].

GANs are highly attractive due to their ability to model the relationship between input and output images for specific domains [16]. They can be trained to produce cloud-free images that closely match the original, clear images of the target domain by employing adversarial loss. Conditional Generative Adversarial Networks (cGANs) have proven to be a valuable method for generating these cloud-free images by learning from paired datasets of both cloudy and clear images [17]. PatchGAN is known for its efficiency and effectiveness in handling images with thin cloud cover while using fewer parameters [18]. On the other hand, a DCGAN-based generative network can successfully reconstruct images with both thin and thick clouds [19]. The integration of GANs with attention mechanisms has recently gained attention, exemplified by the SpA+Edges GAN, which uses a spatial attention map along with edge-GANs to both restore cloud-covered areas and pinpoint regions distorted by clouds [20]. The generator pinpointed the exact heatmap of the overcast regions, while the discriminator used canny edge detectors to find the edges affected by the clouds. The Spatial Attention GAN (SpA GAN) applied both local and global spatial attention to identify and recover clouded areas, demonstrating its effectiveness when tested with the RICE dataset series [21]. Meanwhile, the Haze-Aware Representation Distillation Generative Adversarial Network (HardGAN) achieved single-image dehazing by employing a multi-scale network with a specialized Haze-Aware Representation Distillation layer, and it effectively used a normalization layer to prevent information loss during the reconstruction process [22].

The article explores the difficulties involved in removing clouds from single images using a generative network architecture that incorporates spatial attention mechanisms. Even though GANs are frequently used for content generation in computer vision, they often struggle to produce realistic images, particularly when dealing with genuine remote-sensing data. Creating natural cloud-free images using GANs is more challenging, and it requires incorporating appropriate spatial attention features with convolutional layers to achieve a dependable reconstruction. The key contributions of this proposed work are as follows.

- The MAE-CG model interprets the removal of thin clouds as haze-free reconstruction. It utilizes parallel multi-attention operations incorporating the popular CBAM and CAM architectures to maintain both local and global spatial contexts during the reconstruction process.
- The use of residual connections between short-skip and long-skip additions in the parallel spatial attention frameworks helps preserve important features and reduces the risk of feature loss during training.
- The MAE-CG model was both trained and evaluated using the RICE-1 dataset, and its performance was benchmarked against leading single-image thin cloud removal methods.

The remainder of the article is structured as follows: Section 2 provides a detailed description of the proposed methods, Section 3 covers the experimental procedures and findings, and Section 4 offers a summary along with thoughtful observations and suggestions for future research.

2. PROPOSED METHODOLOGY

2.1. GAN Architecture

The team released a paper on Generative Adversarial Networks (GANs), a sort of machine learning architecture for generative applications [23].GANs have been transformed by number of applications, including image synthesis, super-resolution, data augmentation, image translation, and picture generation, due to their ability to generate extremely realistic data.The two components of a GAN, the generator and the discriminator, are trained simultaneously in an adversarial process.The generator aims to produce data indistinguishable from true data, whereas the discriminator aims to find the difference between generated and genuine data.the generator is trained with random noise as input and generating the data sample as approximate data.The discriminator evaluates data samples to identify authenticity and fraud. Using a GAN, the discriminator teaches the generator to create indistinguishable false data.The characteristics function of a GAN is defined by the minimax equation 1.

$$\min_G \max_D V(D,G) = \mathbb{E}_x[\log D(x)] + \mathbb{E}_y[\log(1 - D(G(y)))] \tag{1}$$

In the above GAN characteristics equation, the generator is indicated by G, discriminator is denoted by D, data from actual distribution $p(x)$ is denoted by x, and data from noisy distribution $p(y)$ is given by y. The minmax GAN problem is defined by the terms $\max_D V(D,G)$ and $\min_G V(D,G)$. $\max_D V(D,G)$ performs the training of D for maximizing the value of mapping function $V(D,G)$ and ensures maximum probability for correctly classifying actual samples while minimizing the probability of misclassifying fake samples. $\min_G V(D,G)$ enable the training of generator G for minimizing the value of mapping function $V(D,G)$ to create phoney samples that are so convincing to the discriminator

that it becomes incapable of telling them apart from genuine ones.

2.2. MAE-CG Architecture

MAE-CG cloud removal consist of parallel multi-attention operations like CBAM and CAM architecture for capturing scene details and overall structure for reconstructing areas hidden by clouds. The architectural details of the generator, discriminator and loss computations are discussed below.

2.2.1. Generator Module

The generator module of the MAE-CG consists of Spatial Feature Extractors (SFE) and a Parallel Attention Module (PAM). The SFEs include multiple convolutional and residual convolutional blocks arranged in a multiscale fashion. Meanwhile, the PAM is designed using a combination of spatial attention maps from the Convolutional Block Attention Module (CBAM) and the Channel Attention Module (CAM). The overall architecture of the generator in the proposed MAE-CG is illustrated in Figure 1.

Initially, a sequence of three 2D convolutional layers processes the input to extract basic feature maps. These feature maps are then passed as input to the generator network. Following this, the features are sent into a residual network consisting of multi-scale convolutional branches. Each branch performs 3 × 3, 5 × 5, and 7 × 7 convolutions to capture information at different scales. The outputs from each convolutional branch are concatenated, resulting in a consolidated multi-scale feature representation. The concatenated features are then processed by a set of three Parallel Attention Modules (PAM), which are designed in a residual addition manner. These modules refine the feature map and extract spatial context and importance. The Parallel spatial attention extractors are developed using CBAM [24] and CAM [25] attention models. This results in a feature map that focuses on essential features while suppressing irrelevant ones. An initial 2D convolution processes the concatenated multi-scale feature map and passes it to the parallel attention architectures of CBAM and CAM. Since CAM is more computationally demanding than CBAM, the CAM branch performs two downsampling operations, computes the CAM attention map, and then performs two upsampling operations to match the spatial resolution of the CBAM-extracted features. Ultimately, both attention maps are concatenated and transmitted to the next layer. The architecture of the proposed PAM module is shown in Figure 2. ReLU activation is applied at each convolutional layer, and the spatial resolution of the feature map is maintained by setting the stride (S) to 1 and using same padding (P) calculated as (k 1)/2, where k is the filter size.

2.2.2. Convolutional Block Attention Module (CBAM)

The CBAM focuses on important spatial and channel areas, highlighting them to improve feature mapping in any traditional CNN. The Channel Attention Module (CAM) and Spatial Attention Module (SAM) work sequentially, with CAM identifying crucial channels and SAM pinpointing relevant spatial locations. The input feature map $\mathcal{F}$ of size $C \times H \times W$ undergoes both average pooling and max pooling in parallel across the channels, resulting in two independent features of size $C \times 1 \times 1$. These features are then processed by a fully connected Multi-Layer Perceptron (MLP) with ReLU activation. The outputs of these transformations are then combined and passed through a sigmoid function (σ) to produce a channel attention map (f_c). The channel attention map f_c is multiplied with $\mathcal{F}$ to produce an intermediate refined feature map $\mathcal{F}'$ of size $C \times H \times W$. The SAM then operates on $\mathcal{F}'$, focusing on the significance of each spatial location across the channels. $\mathcal{F}'$ undergoes both average pooling and max pooling, generating two distinct feature representations. These are combined into a single feature map of size $2 \times H \times W$, which is then processed by a 7×7 convolution kernel (k) and a sigmoid activation function, resulting in a spatial attention map f_s with dimensions $1 \times H \times W$. The representations are combined to create a unified feature map of size $2 \times H \times W$, which is then processed with a 7×7 convolution kernel (k) and a sigmoid activation function to produce a spatial attention map f_s of dimension $1 \times H \times W$. Subsequently, this spatial attention map is multiplied with $\mathcal{F}'$ to generate a final refined feature map $\mathcal{F}''$ with dimensions $C \times H \times W$, incorporating key spatial and channel information. The following equations describe the CBAM operations, where $+$, $\oplus$, $*$, and $\otimes$ represent addition, concatenation, convolution, and multiplication, respectively. The overall architecture of CBAM module shown in Figure 3.

$$f_c = \sigma[MLP(Avg(\mathcal{F})) + MLP(Max(\mathcal{F}))] \quad (2)$$

$$f_s = \sigma[k^{7\times7} * (Avg2D(\mathcal{F}^{'}) \oplus Max2D(\mathcal{F}^{'}))] \quad (3)$$

$$\mathcal{F}^{'} = f_c \otimes \mathcal{F} \quad (4)$$

$$\mathcal{F}^{''} = f_s \otimes \mathcal{F}^{'} \quad (5)$$

2.2.3. Coordinate Attention Module (CAM)

The CAM embeds positional information into channel attention, enabling the network to focus on large, important regions with minimal computational cost. it captures not only cross-channel but also direction-aware and position-sensitive information, which helps models to more accurately locate and recognize the objects of interest.The coordinate attention mechanism consists of two consecutive steps: coordinate information embedding and coordinate attention generation.First, two spatial extents of pooling kernels encode each channel horizontally and vertically. In the second step, a

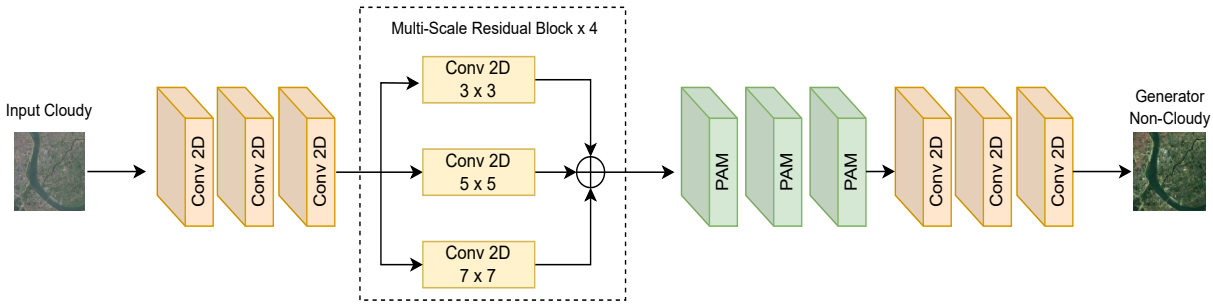

Fig. 1: Overall architecture of the generator module of the proposed MAE-CG

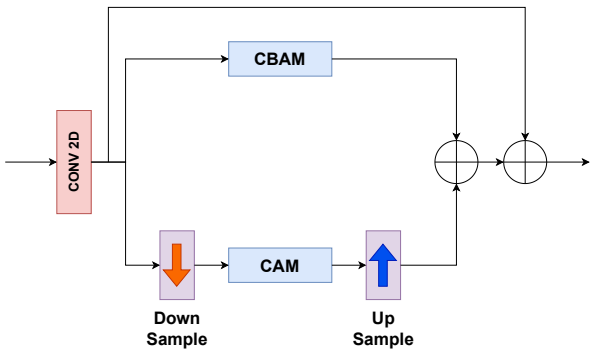

Fig. 2: Architecture of the Parallel Attention Module(PAM)

shared *1 × 1* convolutional transformation function is applied to the concatenated outputs of the two pooling layers. Coordinate attention then splits the resulting tensor into two separate tensors, generating attention vectors for horizontal and vertical coordinates of the input *(X)* along the respective dimensions. The overall architecture of CAM module shown in Figure 4.

$$z^h = \text{GAP}^h(X)$$

$$z^w = \text{GAP}^w(X)$$

$$f = \delta\left(\text{BN}\left(\text{Conv}_1^{1\times1}([z^h; z^w])\right)\right)$$

$$(f^h, f^w) = \text{Split}(f)$$

$$s^h = \sigma\left(\text{Conv}_h^{1\times1}(f^h)\right)$$

$$s^w = \sigma\left(\text{Conv}_w^{1\times1}(f^w)\right)$$

$$Y = X \cdot s^h \cdot s^w$$

where GAP^h and GAP^w denote pooling functions for vertical and horizontal coordinates, and

$$s^h \in \mathbb{R}^{C\times1\times W} \text{ and } s^w \in \mathbb{R}^{C\times H\times1}$$

represent corresponding attention weights.

2.2.4. *Discriminator Module*

The network's discriminator is built using both multiscale and standard convolutions. Similar to the generator module, the resulting cloud-free picture is put into a multiscale convolutional structure after being processed by three 2D convolutional layers. This structure consists of three convolutional branches with kernel sizes of 3 × 3, 5 × 5, and 7 × 7. The outputs from these convolutional branches are concatenated and further processed by a series of 2D convolutional layers. To generate a probability score between 0 and 1, this feature map is first processed by a 2D convolutional layer with an output channel size of 1. After that, it is processed by a sigmoid activation function. If the score is closer to 1, it indicates the discriminator's confidence that the input is authentic; if it is closer to 0, it suggests that the input is fake. To maintain the spatial resolution, the discriminator network also uses appropriate padding-stride settings and ReLU activations. The overall architecture of the discriminator is given in Figure 5.

2.2.5. *Loss Function Design*

The loss function L_{total} for the proposed MAE-CG is a combination of GAN loss (L_{gan}, SSIM Loss (L_{ssim}), and conventional L_1 loss. These combined loss functions usually result in high-quality, realistic, generative outcomes due to their combined effects in improving the training process with regularization. The overall loss L_{total} is given by the Equation 6.

$$L_{total} = L_{gan} + L_{ssim} + L_1 \tag{6}$$

L_{gan} is the combination of generator loss L_G, and discriminator loss LD, The L_G and L_D is given by Equation 7, and the summed L_{gan} is given by Equation 8.

$$\begin{aligned} L_G &= -\mathbb{E}_y[\log(1 - D(G(y))] \\ L_D &= \mathbb{E}_x[\log D(x)] - \mathbb{E}_y[\log(1 - D(G(y)))] \end{aligned} \tag{7}$$

$$L_{gan} = \mathbb{E}_x[\log D(x)] + \mathbb{E}_y[\log(1 - D(G(y)))] \tag{8}$$

The L_{ssim} is defined as a metric that quantifies the quality of the generated image and original in terms of structural information, luminance, and contrast. The proposed loss function

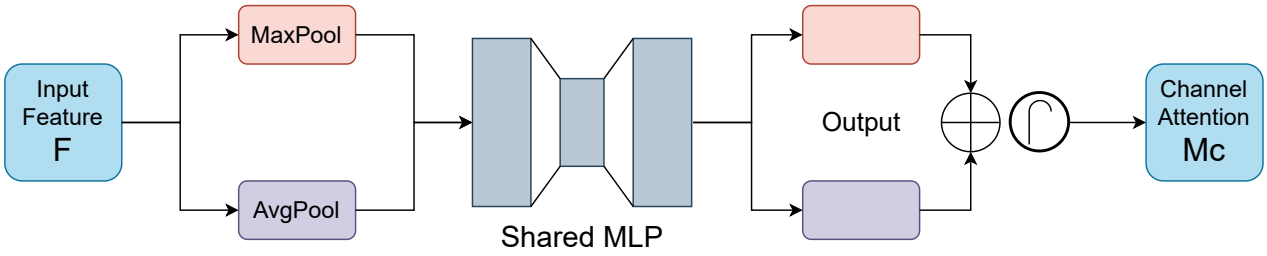

Fig. 3: Overall architecture of CBAM

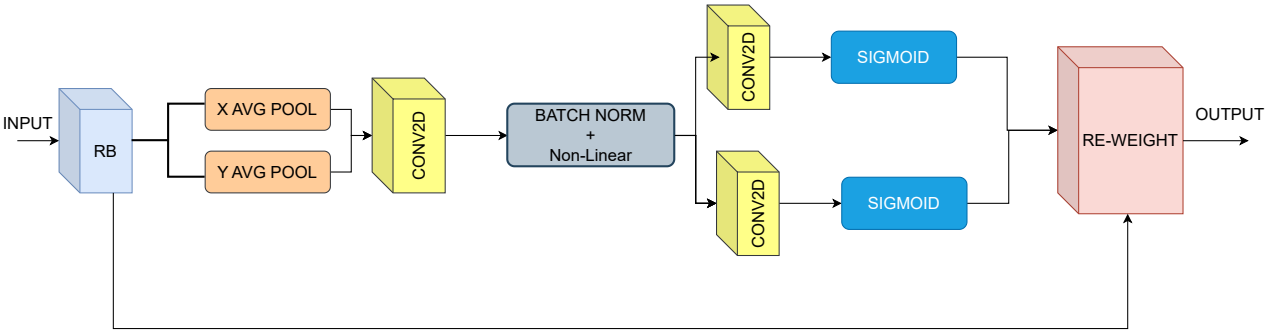

Fig. 4: Overall architecture of CAM

for MAE-CG incorporates L_{ssim} to preserve key features and details that might be perceptually important, leading to realism in generated images. The structural similarity between the generated image y and actual image x is given by Equation 9, where μ, σ, σ^2, and C indicate mean, variance, covariance and stabilization constant respectively.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{9}$$

The L_{ssim} loss between the real and generated data is defined by the following Equation 11

$$L_{ssim} = 1 - SSIM(real, generated) \tag{10}$$

The final component of the L_{total} is the standard L_1 loss computed between the real and generated data. The L_1 loss act as a regularization component and brings generalization to the network being trained. The L_1 loss between the real and generated image is given by the Equation **??**, where N is the total number of pixels in the image.

$$L_1(real, generated) = \frac{1}{N}\sum_{i=1}^{N} \|real_i - generated_i\|_1 \tag{11}$$

3. RESULTS AND DISCUSSIONS

3.1. Experimental Setup

The RICE-1 dataset is a baseline cloud/mist dataset designed to improve satellite imagery by removing clouds and mist to reconstruct the underlying surface [26]. The proposed MAE-CG network is trained and tested using the RICE-1 dataset, which contains 500 pairs of images, each with a spatial resolution of 512 × 512 pixels, both with and without clouds. The RICE-1 dataset is curated from the Google Earth Platform. Training of the MAE-CG model is conducted on the AI computing facility at NIT Puducherry, using an NVIDIA DGX A100 GPU server with 320GB of memory. The model's performance is evaluated using quantitative metrics, including PSNR, SSIM, MSE, and RMSE. A sample from the RICE-1 dataset is shown in Figure 4.

3.2. Result Analysis

The efficiency of the MAE-CG architecture is evaluated against four different GAN-based haze removal techniques: DHI [27], AOD-Net [28], DehazeNet [29], and Cycle-Dehaze [30]. Figure 6 displays the total loss recorded during both the training and validation phases of model development. Training and validation enhance the model's reliability and generalizability. As training progressed through more epochs, the validation loss decreased and better aligned with the increased training accuracy. Training and validation losses are

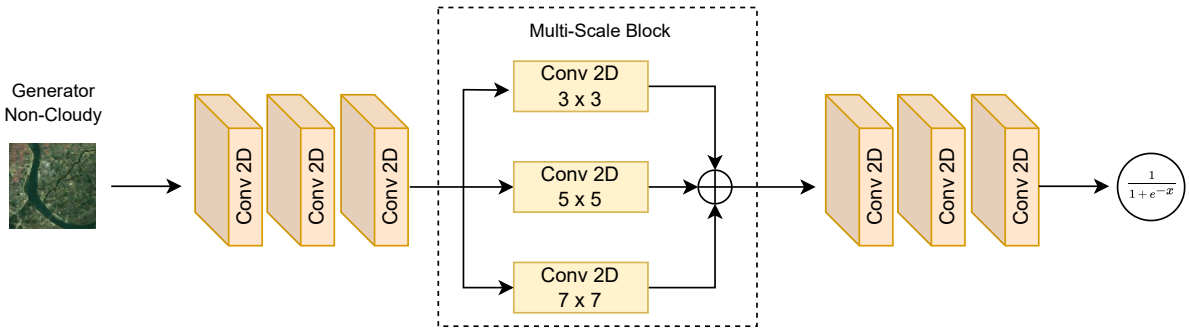

Fig. 5: Overall architecture of the discriminator module of the proposed MAE-CG

Fig. 6: Sample images from RICE-1 dataset columns of (a), (b), (c), and (d) represents thin cloud image and their corresponding non-cloudy counterpart

compared at each epoch, and the model with the smallest difference between these losses is selected as the best model and saved. Table 1 presents the model's performance comparisons, and Figure 8 provides a visual representation of the MAE-CG performance compared to SOTA techniques.

Table 1: Models performance comparison against SOTA dehaze architectures

Models	PSNR	SSIM	MSE	RMSE
DHI	15.01	0.805	0.24	0.48
AOD-Net	18.76	0.814	0.36	0.60
DehazeNet	23.14	0.838	0.15	0.38
Cycle-Dehaze	23.73	0.851	0.22	0.46
Proposed	24.80	0.862	0.16	0.40

Table 1 presents a relative analysis metrics of the performance of state-of-the-art (SOTA) cloud removal techniques using four key metrics: PSNR, SSIM, MSE, and RMSE. PSNR (Peak Signal-to-Noise Ratio) Evaluates image quality, with increased values signifying improved quality. SSIM (Structural Similarity Index)extends from 0 to 1, where increased values represent better structural similarity.MSE (Mean Squared Error) and RMSE (Root Mean Squared Error) measure error, with lower values signifying higher quality for both metrics.The recommended MAE-CG model has better performance compared to other techniques in terms of PSNR and SSIM.The model acquired the highest PSNR value of 24.80, Showing better reconstruction quality,it also showcased the highest SSIM score of 0.862 Indicates a high degree of structural similarity to the true data.The MAE-CG model achieved the highest scores for these parameters, but DehazeNet outperformed it in terms of MSE and RMSE.Even so, the 0.01 MSE difference can be ignored due to the superior PSNR and SSIM.DehazeNet excels in minimizing errors, as shown by its lowest MSE value of 0.15 and RMSE value of 0.38. These figures indicate that DehazeNet has the smallest average and root-mean-squared deviations from the reference images. Even so, its performance on the PSNR and SSIM evaluation metrics was minimal. Cycle-Dehaze and AOD-Net illustrate average performance. Cycle-Dehaze recorded higher PSNR and SSIM values than AOD-Net but also had slightly higher error values. In general, the MAE-CG model is excellent at creating realistic and well-structured reconstructions. It is important to note that although DehazeNet is effective at minimizing forecast errors, it does not achieve high scores for quality and structural similarity in its reconstructions.

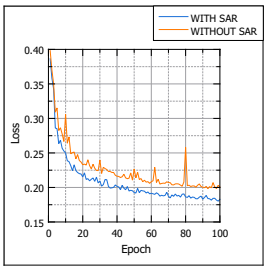

Fig. 7: Training and validation loss curves for the suggested MAE-CG

4. CONCLUSIONS

Effective cloud removal is essential for improving the quality of aerial or satellite photography, as airborne optical data is crucial in many geospatial applications, including disaster management, agricultural evaluation, and environmental monitoring. Strong and dependable thin cloud removal techniques can enhance the confidence and precision of these applications.

The main goal of this research is to design and develop a robust, generalizable cloud removal model. The proposed Multi Attention Enhanced Thin Cloud-Removal Generative Adversarial Network (MAE-CG) demonstrated superior performance across most metrics when tested against the RICE-1 dataset. The MAE-CG model integrates two sophisticated spatial-attention mechanisms: the Convolutional Block Attention Module (CBAM) and the Coordinative Attention Module (CAM). These modules enhance the model's ability to effectively extract superior spatial context, leading to high-quality, cloud-free image reconstruction.

The MAE-CG model achieved the highest Peak Signal-to-Noise Ratio (PSNR) value of 24.80 and the highest Structural Similarity Index (SSIM) score of 0.862. These metrics indicate that it produces high-quality images with the best structural similarity to the reference images. The results highlight the importance of retaining image integrity and quality while efficiently removing thin clouds.

Future studies should focus on enhancing the current methods used for this task, such as utilizing recently developed cloud detection technologies, machine learning algorithms informed by physics, and advancements in computational capabilities. These improvements aim to increase the overall usefulness and effectiveness of cloud removal models. Further research may also explore how well these models perform in other contexts and datasets. **Pending**

CBAM
CAM
Train Validation Loss Curve Original
Actual Result Images

5. REFERENCES

[1] K Prinsa and E Saritha, "Detection and removal of clouds on remote sensing images," in *2018 4th International Conference for Convergence in Technology (I2CT)*, 2018, pp. 1–5.

[2] Daoyu Lin, Guangluan Xu, Xiaoke Wang, Yang Wang, Xian Sun, and Kun Fu, "A remote sensing image dataset for cloud removal," *CoRR*, vol. abs/1901.00600, 2019.

[3] Bo Jiang, Haozhan Chong, Zhenyu Tan, Hang An, Haoran Yin, Shengmei Chen, Yanchao Yin, and Xiaoxuan Chen, "Fdt-net: Deep-learning network for thin-cloud removal in remote sensing image using frequency-domain training strategy," *IEEE Geoscience and Remote Sensing Letters*, vol. 21, pp. 1–5, 2024.

[4] Zhaocong Wu, Jun Li, Yisong Wang, Zhongwen Hu, and Matthieu Molinier, "Self-attentive generative adversarial network for cloud detection in high resolution remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 10, pp. 1792–1796, 2020.

[5] Suphongsa Khetkeeree, Bannakorn Petchthaweetham, Sompong Liangrocapart, and Sanun Srisuk, "Sentinel-2 image dehazing using correlation between visible and infrared bands," in *2020 8th International Electrical Engineering Congress (iEECON)*, 2020, pp. 1–4.

[6] Takahiro Toizumi, Simone Zini, Kazutoshi Sagi, Eiji Kaneko, Masato Tsukada, and Raimondo Schettini, "Artifact-free thin cloud removal using gans," in *2019 IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 3596–3600.

[7] Yue Zi, Fengying Xie, Ning Zhang, Zhiguo Jiang, Wentao Zhu, and Haopeng Zhang, "Thin cloud removal for multispectral remote sensing images using convolutional neural networks combined with an imaging model," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 3811–3823, 2021.

[8] Yue Gao, Yong Wang, Haitao Lv, and Jiang Qian, "A revised rtm-based algorithm to remove thin clouds within visible band data of sentinel-2a," in *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 1418–1421.

[9] Jie Kong, GenSheng Hu, and Dong Liang, "Thin cloud removing approach of color remote sensing image based on support vector machine," in *2010 Asia-Pacific Conference on Wearable Computing Systems*, 2010, pp. 131–135.

[10] Shuli Chen, Xuehong Chen, Jin Chen, Pengfei Jia, Xin Cao, and Canyou Liu, "An iterative haze optimized transformation for automatic cloud/haze detection of landsat imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 5, pp. 2682–2694, 2016.

[11] Weikang Yu, Xiaokang Zhang, Man-On Pun, and Ming Liu, "A hybrid model-based and data-driven approach for cloud removal in satellite imagery using multi-scale distortion-aware networks," in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, 2021, pp. 7160–7163.

[12] Yujun Guo, Wei He, Yu Xia, and Hongyan Zhang, "Blind single-image-based thin cloud removal using a cloud perception integrated fast fourier convolutional

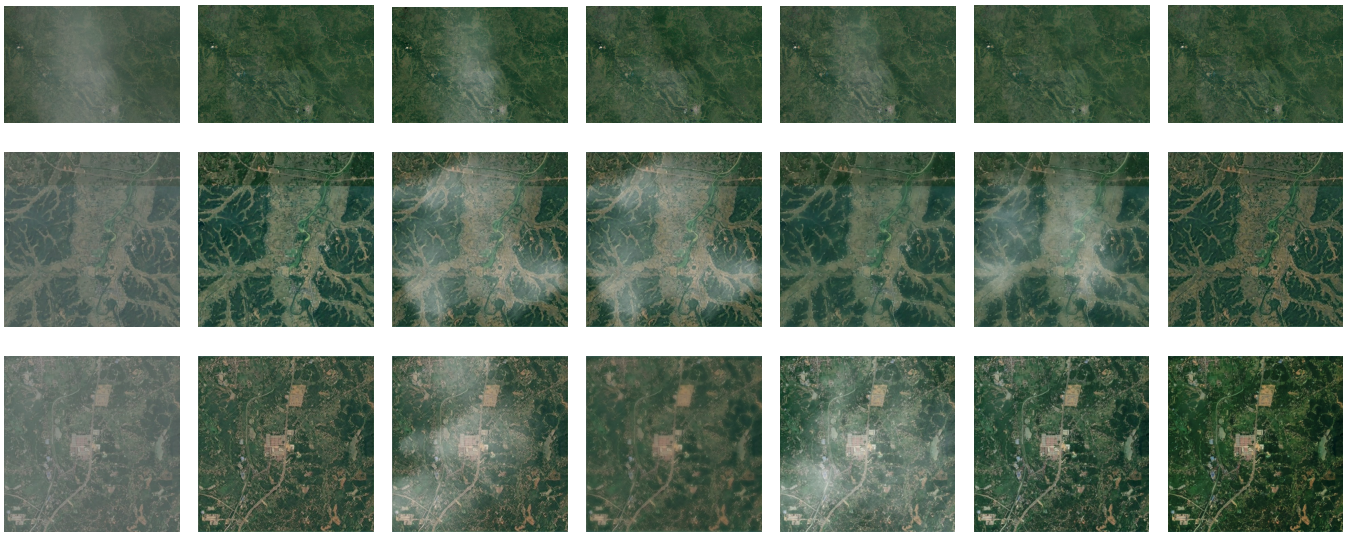

Fig. 8: Subjective comparison of the proposed MAE-CG against SOTA techniques, (a) Hazy Image, (b) Ground-Truth, (c) DHI, (d) AOD-Net, (e) DehazeNet, (f) Cycle-Dehaze, (g) Proposed MAE-CG

network," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 206, pp. 63–86, 2023.

[13] Bo Jiang, Haozhan Chong, Zhenyu Tan, Hang An, Haoran Yin, Shengmei Chen, Yanchao Yin, and Xiaoxuan Chen, "Fdt-net: Deep-learning network for thin-cloud removal in remote sensing image using frequency-domain training strategy," *IEEE Geoscience and Remote Sensing Letters*, vol. 21, pp. 1–5, 2024.

[14] Chull Hwan Song, Hye Joo Han, and Yannis Avrithis, "All the attention you need: Global-local, spatial-channel attention for image retrieval," *CoRR*, vol. abs/2107.08000, 2021.

[15] Qiaoqiao Yang, Guangxing Wang, Yaxuan Zhao, Xiaoyu Zhang, Guoshuai Dong, and Peng Ren, "Multi-scale deep residual learning for cloud removal," in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020, pp. 4967–4970.

[16] Praveer Singh and Nikos Komodakis, "Cloud-gan: Cloud removal for sentinel-2 imagery using a cyclic consistent generative adversarial networks," in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, 2018, pp. 1772–1775.

[17] Mehdi Mirza and Simon Osindero, "Conditional generative adversarial nets," 2014.

[18] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros, "Image-to-image translation with conditional adversarial networks," 2018.

[19] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016.

[20] Amal S Namboodiri, Rakesh Kumar Sanodiya, and PV Arun, "Remote sensing cloud removal using a combination of spatial attention and edge detection," in *2023 11th International Symposium on Electronic Systems Devices and Computing (ESDC)*, 2023, vol. 1, pp. 1–6.

[21] Heng Pan, "Cloud removal for remote sensing imagery via spatial attention generative adversarial network," 2020.

[22] Qili Deng, Ziling Huang, Chung-Chi Tsai, and Chia-Wen Lin, "Hardgan: A haze-aware representation distillation gan for single image dehazing," in *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, Eds., Cham, 2020, pp. 722–738, Springer International Publishing.

[23] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, Eds. 2014, vol. 27, Curran Associates, Inc.

[24] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon, "CBAM: convolutional block attention module," *CoRR*, vol. abs/1807.06521, 2018.

[25] Qibin Hou, Daquan Zhou, and Jiashi Feng, "Coordinate attention for efficient mobile network design," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 13713–13722.

[26] Daoyu Lin, Guangluan Xu, Xiaoke Wang, Yang Wang, Xian Sun, and Kun Fu, "A remote sensing image dataset for cloud removal," *CoRR*, vol. abs/1901.00600, 2019.

[27] Xiaoxi Pan, Fengying Xie, Zhiguo Jiang, and Jihao Yin, "Haze removal for a single remote sensing image based on deformed haze imaging model," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1806–1810, 2015.

[28] Boyi Li, Xiulian Peng, Zhangyang Wang, Jizheng Xu, and Dan Feng, "Aod-net: All-in-one dehazing network," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4780–4788.

[29] Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao, "Dehazenet: An end-to-end system for single image haze removal," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.

[30] Deniz Engin, Anil Genc, and Hazim Kemal Ekenel, "Cycle-dehaze: Enhanced cyclegan for single image dehazing," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 938–9388.