CLOUDTRAN: CLOUD REMOVAL FROM MULTITEMPORAL SATELLITE IMAGES USING AXIAL TRANSFORMER NETWORKS

Dionysis Christopoulos*, Valsamis Ntouskos, Konstantinos Karantzalos

Remote Sensing Lab, National Technical University of Athens, Greece -
christopoulosdion@mail.ntua.gr, ntouskos@mail.ntua.gr, karank@central.ntua.gr

KEY WORDS: cloud-free, cloud detection, reconstruction, reflectance, time series, autoregressive models, sentinel-2

ABSTRACT:

We present a method for cloud-removal from satellite images using axial transformer networks. The method considers a set of multitemporal images in a given region of interest together with the corresponding cloud masks, and delivers a cloud-free image for a specific day of the year. We propose the combination of an encoder-decoder model employing axial attention layers for the estimation of the low-resolution cloud-free image, together with a fully parallel upsampler that reconstructs the image at full resolution. The method is compared with various baselines and state-of-the-art methods on two Sentinel-2 datasets, showing significant improvements across multiple standard metrics used for image quality assessment.

1. INTRODUCTION

Cloud removal from satellite images is a crucial part of remote sensing tasks, especially for the production of composite mosaics covering a region of interest and the analysis of multitemporal data, including, among others, change detection (Liu et al., 2019, Papadomanolaki et al., 2021) and detection of phenological events (Karakizi et al., 2018, Franchetti et al., 2019). Nowadays, earth observation data of high spatial and temporal resolution are available thanks to the commissioning of a large fleet of satellites continuously monitoring the earth. However, a large proportion of the collected data cannot be used as they are affected by clouds. This proportion depends on the geographic location of the region, its climate characteristics, and the season when acquisition takes place.

Cloud removal methods can be divided into two main categories, namely, those taking advantage of the temporal evolution of the pixel values and those attempting inpainting/gap-filling of parts affected by clouds in a single image, with the former having the advantage of being conditioned on the history of each pixel, reducing the prediction uncertainty for the missing values. Deep learning methods have been developed for both these categories. For single image cloud removal, the proposed methods typically employ generative adversarial neural networks (Singh and Komodakis, 2018, Pan, 2020). Multitemporal cloud removal methods on the other hand typically employ conditional generative models, as ResNet and U-Net, enriched with (ConvLSTM) modules (e.g. (Ebel et al., 2022, Sebastianelli et al., 2020)). As these methods capture both the temporal and spatial relations of the pixels, although more challenging and computationally intensive, they typically lead to significant improvements in the produced cloud-free images.

Transformer networks have pushed the state-of-the-art in numerous natural language processing (Vaswani et al., 2017), computer vision (Dosovitskiy et al., 2020), and remote sensing tasks (Bazi et al., 2021). However, their application in image generation and especially in the multitemporal case is quite challenging due to their quadratic dependence on the input size (i.e. number of pixels). We propose a novel method for cloud removal from multitemporal satellite images based on transformer networks (Vaswani et al., 2017) that use the axial attention mechanism. Axial attention (Ho et al., 2019) significantly improves temporal and parameter efficiency by applying attention independently on each tensor axis, without sacrificing the model's receptive field.

* Corresponding author: christopoulosdion@mail.ntua.gr

2. RELATED WORK

Cloud removal methods can be divided into two main categories, based on whether they use context from the temporal evolution of the pixel values, like ours, or they attempt to replace the cloudy part with meaningful content in a single image.

Single image cloud removal methods Cloud-GAN (Singh and Komodakis, 2018) is a generative adversarial network which uses two generator and two discriminator networks. By employing a cycle consistency loss, the generator is restrained to map the input domain to target and then back to the input domain producing an output that is as close as possible to the original input. This method does not require a paired dataset with cloudy and cloud-free images of the same region nor any extra sources such as SAR data.

SpA-GAN (Pan, 2020) utilizes a Generator called spatial attentive network (SPANet) and a Discriminator which is a standard convolutional neural network. The generator employs a spatial attention mechanism with a local-to-global perspective to detect cloudy regions and better capture the relative context as to produce results with higher fidelity.

SACTNet (Liu and Hu, 2021) consists of two networks, first a transformer-based network with a content extractor to get context from the ground-truth and the cloudy image, a correlation embedding and a soft attention module to synthesize texture elements. Second, a backbone generative adversarial network that utilizes a spatial attention mechanism in a recurrent style in order to obtain spatial information for the regions affected by clouds. This method gives state-of-the-art results in the RICE dataset (Lin et al., 2012) for thin and thick cloud removal tasks.

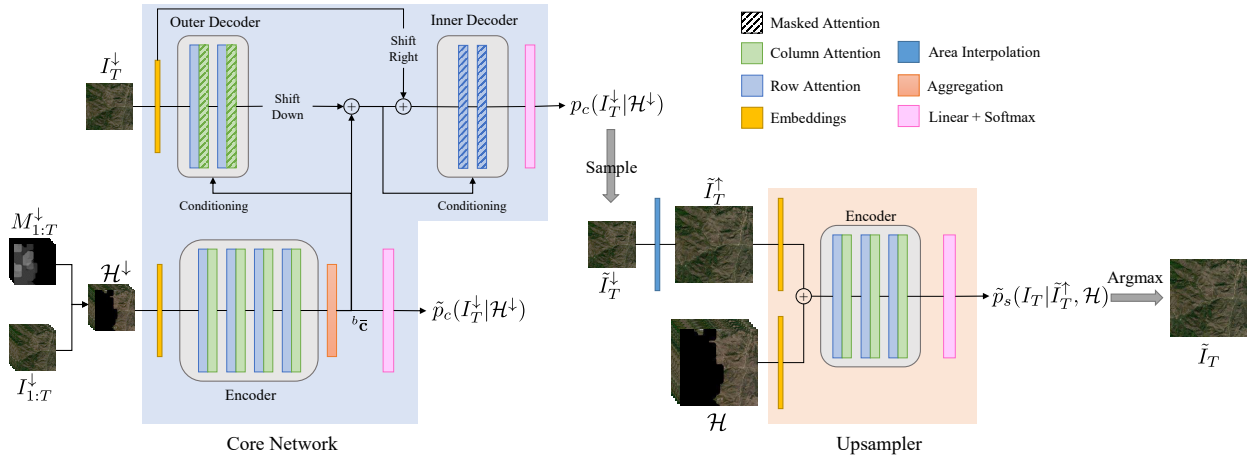

Figure 1. Proposed architecture.

Multi-temporal cloud removal methods The work introducing the SEN12MS-CR-TS multi-modal multi-temporal dataset (Skakun et al., 2022), proposes also two multi-temporal cloud removal methods. The first, is based on a 3D Convolution Neural Network that considers time-series of Sentinel-1 (SAR) and Sentinel-2 (optical) data to produce a single cloud-free image. The second is a sequence-to-sequence method, with a 3D Encoder-Decoder architecture in a U-Net style, that takes as input time-series of Sentinel-1 images and produces the corresponding time-series of Sentinel-2 images.

In (Sebastianelli et al., 2020) a conditional Generative Adversarial Network (cGAN) and a Convolutional Long Short-Term Memory (convLSTM) are employed to extract spatio-temporal features from multitemporal Sentinel-1 (SAR) and Sentinel-2 (optical) data respectively. Finally, the extracted features are fused by a U-shaped convolutional neural network to predict the final output.

In this work, instead, we propose a method employing transformer networks for reconstructing a cloud-free target image considering only a small number of previously acquired optical images of the same region.

Video inpainting Video inpainting is a closely related computer vision task. (Chang et al., 2019) propose the Learnable Gated Temporal Shift Module (LGTSM) which handles masked videos with 2D convolutions. This method models the spatio-temporal features, processing context from a variable number of temporal neighbors.

In (Liu et al., 2021) transformer networks are employed for video inpainting via fine-grained feature fusion. Two task-specific modules are introduced which are used in tokenization and de-tokenization before and after Transformer layers, enabling sub-patch level information interaction for addressing blurring in masked regions caused by patch splitting. In our work, by using axial attention, we avoid the problems related to patch splitting based vision transformer networks.

3. METHOD

We aim at producing a cloud-free version of a target image I_T given a stack of cloud-affected images $I_{1:T}$ for a certain temporal horizon T. Our method is based on the use of transformer networks (Vaswani et al., 2017) which are powerful learning based estimators. Transformer networks are autoregressive models, i.e. each sequence element they produce, apart from the input, is also conditioned on all previously generated outputs.

Considering that both the number of parameters and their computational complexity depends quadratically on the number of sequence elements, for images, where sequences of pixels are considered, transformer networks can have prohibitively large model sizes. To address this issue, several solutions have been proposed ((Han et al., 2020) provides a survey). In this work, we consider Axial Transformers whose basic building block is axial attention (Ho et al., 2019). An axial attention block performs self-attention over a single axis (here columns and rows of an image), mixing information along that axis while keeping it independent along the other axes. This helps to reduce the model complexity to $\mathcal{O}(n\sqrt{n})$, where n is the total number of pixels, providing $\mathcal{O}(\sqrt{n})$ savings. Models employing axial attention can still capture global receptive field by combining multiple axis attention blocks spanning different axes. The resulting autoregressive network models a distribution $p(x_{i,j})$ over a pixel x at position (i, j) by processing all the past context from $x_{i,<j}$ and $x_{<i}$ following the raster scan order. Each axial attention block is composed of a self-attention block passing through a feed forward block consisting of a layer normalization and a two layers network.To prevent the model from considering ”future” outputs during training, these outputs are masked out, using a masked axial attention block variant.

A full axial-transformer architecture is composed of an encoder, capturing context from individual channels or images, an outer decoder capturing context of entire rows, and an inner decoder considering context within a single row. Specifically, the encoder consists of unmasked row and column attention layers and makes each pixel xi, j depend on all the previous channels or images. The output of the encoder is used as context to condition the decoder. In this work, we follow the conditioning approach proposed in (Kumar et al., 2021).

Regarding the decoder, its outer part consists of unmasked row and masked column attention layers and makes each pixel $x_{i,j}$ depend on all the previous rows $x_{<i}$. The output context is then shifted down by a single row in order to ensure that it contains information only from previous rows and not from its own. This context is then summed with the encoder context and used to condition the inner decoder. The inner decoder consists

of masked row attention layers, capturing information from the previous pixels of the same row $x_{i,<j}$. The inner decoder embeddings are shifted right by one pixel, ensuring that the current pixel is excluded from the receptive field. The new output context is then passed through a final dense layer to produce logits of shape $H \times W \times V$, where V corresponds to the range of pixel values at each location.

Outputs of autoregressive models are produced by sampling a single pixel at a time, which is a particularly computationally expensive process as the whole network needs to be reevaluated each time. Axial transformers support semi-parallel autoregressive sampling where the encoder runs once per image, the outer decoder once per-row and the inner decoder once per-pixel. The context from the encoder and outer decoder conditions the inner decoder, which generates a row, pixel-by-pixel. After generating all pixels in a row, the outer decoder runs to recompute context and condition the inner decoder, to generate the next row. Finally, after all the pixels of an image are generated, the encoder recomputes context in order to generate the next image.

3.1 Proposed architecture

The complete architecture of the proposed CloudTran method is presented in Figure 1. While the efficiency gains achieved by using axial attention blocks are substantial, it is still challenging to build encoder-decoder models for images of increased resolution (e.g. 256×256) as, besides increased model size, sampling becomes excessively slow for generating images with higher resolutions. To address this issue, following similar ideas from (Menick and Kalchbrenner, 2019) and (Kumar et al., 2021), we split the cloud removal problem into two sub-problems, each addressed by a specialized network. The first network (core) is an encoder-decoder model that performs cloud removal to a downsampled version of the original inputs, while the second one (upsampler) brings the output of the core network to the original resolution.

Specifically, the core network takes as input a stack of downsampled T image patches $I_t^{\downarrow}$ and produces a cloud-free version of the downsampled target image $I_T^{\downarrow}$. The encoder, comprised of four layers of row and column attention, processes the input tensor $\mathcal{H}^{\downarrow} \in \mathbf{R}^{H^{\downarrow} \times W^{\downarrow} \times B \times T}$ made of T image patches corresponding to consecutive dates, with $H^{\downarrow} \times W^{\downarrow}$ the size of the downsampled images and B the number of bands considered. In each patch I_t, cloudy regions are masked out using image masks $\mathcal{M}^{\downarrow}$. The encoder produces separate contexts $\mathbf{c}_t$ for each date which are subsequently aggregated, producing a single context $\bar{\mathbf{c}}$ for each band. The aggregated context $\bar{\mathbf{c}}$ is then used for conditioning the layers of the decoder whose output captures the per-pixel distribution over the admissible values of the downsampled cloud-free target image, conditioned on the input tensor $\mathcal{H}^{\downarrow}$, namely:

$$p_c(I_T^{\downarrow}|\mathcal{H}^{\downarrow}) = \prod_i \prod_j p_c\left(I_T^{\downarrow}(i,j)|I_T^{\downarrow}(<i,\cdot), I_T^{\downarrow}(i,<j), \mathcal{H}^{\downarrow}\right).$$

The context is considered independently for each band $b \in B$ of the input tensor. The model distinguishes between contexts corresponding to different bands via positional encoding. This increases the flexibility of the model with respect to the number of bands of the input tensor.

As proposed in (Kumar et al., 2021), to increase stability of the training process we also model the per-pixel distribution from the encoder output $\tilde{p}_c(I_T^{\downarrow}|\mathcal{H}^{\downarrow})$, by adding a dense and a softmax layer after the encoder's aggregation layer.

The upsampler network is a parallel model, i.e. all outputs are produced at once, given the input context. It is composed of three layers of row and column attention, and captures the per-pixel distribution of the cloud-free target image given the input tensor and the bilinearly upsampled cloud-free image, namely $p_s(I_T|\tilde{I}_T^{\uparrow}, \mathcal{H})$.

Each network is trained independently by minimizing the negative log-likelihood of the data, considering the cloud-free version of the last image I_T^{GT}, which in the case of the core network, is also given as input to the decoder during training. During inference, to generate the low-resolution cloud-free image $\tilde{I}_T^{\downarrow}$, the context from the input data tensor is computed by the encoder and each pixel is sampled from decoder in an autoregressive fashion. We make use of the semi-parallel sampling property of axial transformers to speed up the process, which avoids reevaluating the entire network for each pixel of the generated image. As the model considers the context corresponding to each band separately, sampling of each band is performed independently and the target image is obtained by stacking together the sampled bands. The image generated from the core network is then passed to the upsampler to produce the target cloud-free image $\tilde{I}_T$.

4. EXPERIMENTAL EVALUATION

4.1 Datasets

In-house Multi-temporal Dataset We consider an in-house dataset consisting of fifty-six (56) Level-2A (L2A) products of Sentinel-2 satellite images, corresponding to different Days of Year (DOYs) in the period 2018-2019. Level-2A products provide Bottom of Atmosphere (BOA) reflectance images derived from the associated Level-1C products. Each product is a $100 \times 100\ km^2$ tile. We consider the bands B02, B03, B04 with $10m$ spatial resolution and create multiband RGB tiles for each of the 56 different dates, from which we cut 512×512 patches producing 441 multi-temporal patches of dimensions $512 \times 512 \times 56$ in total. We compute cloud masks by considering all pixels with non-zero value in the CLD band that corresponds to each L2A product. We select a 5×5 window of 512×512 patches from every corner of the tile for testing/validation. This translates to a total of 100 regions ($\sim 20\%$) used for testing/validation, while the training set consists of the remaining 341 regions ($\sim 80\%$).

SEN12MS-CR-TS We also consider the public SEN12MS-CR-TS dataset (Ebel et al., 2022) to further validate our proposed architecture. This multi-modal and multi-temporal dataset contains radar and optical observations collected via Sentinel-1 and Sentinel-2 satellites for 53 ROIs (Regions Of Interest) worldwide[1], with 30 time samples provided for each patch-wise 256×256 observation. In this work we consider the bands B02, B03, B04 of Sentinel-2 images (Level 1C top-of-atmosphere reflectance products) to create 12293 multitemporal RGB patches in total. We follow the train/test splits indicated in (Ebel et al., 2022), resulting in 1031 testing/validation and 11262 training patches.

[1] although only 43 were available in the published dataset at the time we accessed it

We produce cloud masks using the s2cloudless cloud detector (Skakun et al., 2022), which takes as input the S2 image with its original 13 bands and returns the binary raster cloud mask, where 0 indicates pixels classified as clear-sky while 1 indicates pixels classified as clouds. Before running the s2cloudless detector, the raw reflectance values are divided with the quantification value 10000 for every band.

4.2 Implementation

We train the proposed CloudTran model on a Workstation equipped with two NVIDIA Quadro RTX 6000 GPUs with 24GB of VRAM each. Unless explicitly stated otherwise, each model of the core transformer network has been trained for 15000 iterations with a batch size equal to one, using the RMSProp algorithm with a learning rate equal to $3 \cdot 10^{-4}$, and by taking the exponential moving average of the parameters during training with a decay value equal to 0.999. The same optimization parameters were used also for training the spatial upsampler. The relative weight between the encoder and the decoder log-likelihood losses of the core network is taken equal to 0.99.

The input tensor $\mathcal{H}$ is formed using patches corresponding to T consecutive dates while considering the last one I_T as the target image. For training and validation, we consider T images captured during the summer period (DOY 55 for our dataset and 15 for SEN12MS-CR-TS), to maximize the number of patches having a cloud-free target image, i.e. with a cloud coverage below 5%. For our dataset, the input patches are randomly cropped to 256×256 for training and centrally cropped to the same dimension for evaluation. The SEN12MS-CR-TS dataset contains patches of 256×256 and we define the input size for training and evaluation as 224×224. For testing, we consider from within the testing/validation split, dates in the spring period, and require that the target image has a cloud coverage above 5%.

Regarding the cloud masks, for each dataset we build a dictionary of masks with moderate coverage (typically $5 - 30\%$ coverage) $\mathcal{M}$, from all the patches of the dataset split considered. To increase the variety of masks that the models encounter during training, after masking out the clouds in each patch, masks from $\mathcal{M}$ are applied randomly to each patch of the input tensor $\mathcal{H}$, before these are fed to the network. For validation, each patch is masked using the corresponding cloud masks, and a random mask from the dictionary is applied to the target image.

The inputs of the core model are first downsampled to 64×64. During training, the clean target image I_T is also provided as input to the decoder. For training the spatial upsampler, the target image is first downsampled and then bilinearly upsampled back to the original resolution, and provided to the model concatenated with the input tensor $\mathcal{H}$. Supervision to both models is provided via the clean target image. During inference, the input patches, masked with the corresponding cloud masks, are provided to the core model. The low-resolution cloud-free image is produced by sampling the model in an autoregressive fashion. This image is then provided to the spatial upsampler after bilinear upsampling, together with the input tensor $\mathcal{H}$. The pixel values of the cloud-free image are taken as the maximum likelihood estimates of the output distribution.

4.3 Ablation

We perform several ablations on our dataset both for the core and the upsampler models to assess the contribution of different parameter choices on the quality of the reconstructed patches. Ablation results are compared with respect to the validation loss of the models, which is measured in bits per dimension (bits/dim) (Papamakarios et al., 2017).

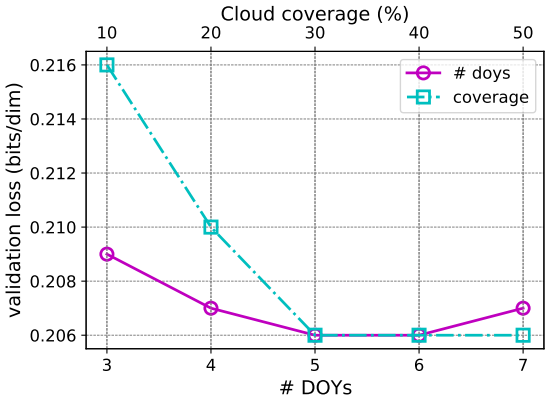

Figure 2. Validation loss in relation to number of DOYs provided as input and maximum cloud coverage considered during training.

First, we study the effectiveness of the context aggregation method. The results are presented in Table 1, where *Sum* corresponds to the depth-wise addition of the per-DOY contexts, *Attention* corresponds to the use of an attention block operating in the third axis of the context tensor, and *Convolution* corresponds to the use of 1×1 convolution on the context tensor. It is evident that simple tensor reduction is not able to produce a single context summarizing sufficiently well the whole tensor. Learnable blocks on the other hand perform much better, reducing the validation loss almost by an order of magnitude. Between attention and convolution, the latter performs slightly better, while employing fewer parameters. Hence, unless otherwise stated, for the experiments that follow we consider the 1×1 convolution as the preferred context aggregation mechanism.

Context Aggregation	Validation Loss
Sum	1.273
Attention	0.239
Convolution	0.206

Table 1. Ablation on context aggregation method.

We also consider the impact of the model size on the final result. In particular, Table 2 reports the validation loss for core models of size 64, 128, 256, and 512. The model size, defines the common size used for the input embedding, as well as the size of the attention block and the feed forward block. Although, increasing the core model size from 64 to 128 improves the reconstruction quality, further increasing the model size to 256 and 512, leads to significant degradation. We justify this due to the large number of model parameters and the relatively small size of the dataset employed.

Model Size	Params	Validation Loss
64	0.9M	0.224
128	3.2M	0.206
256	12.0M	3.44
512	46.4M	5.54

Table 2. Ablation on core model size.

Table 3, reports the validation loss for spatial upsampler models

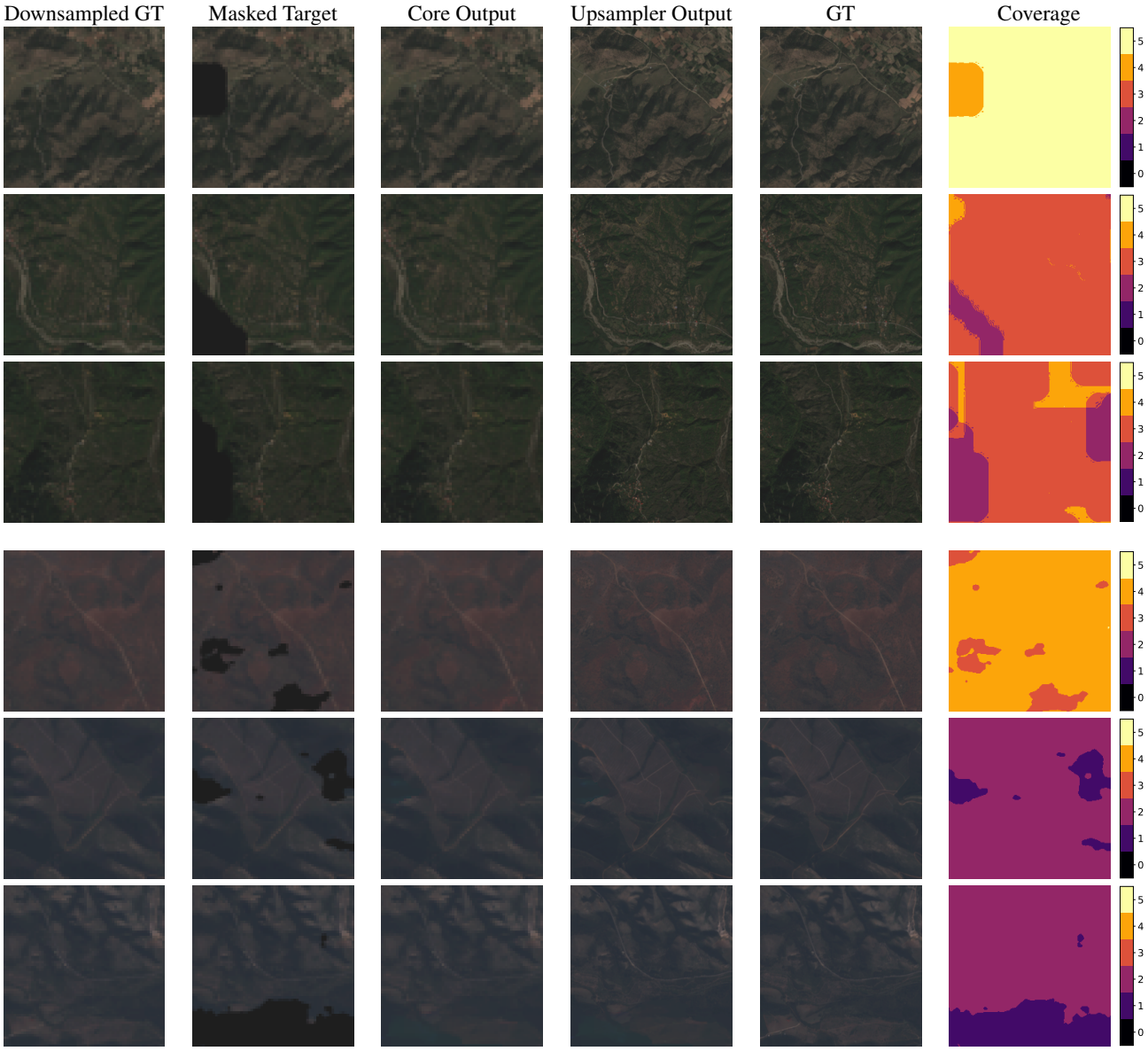

Figure 3. Random samples from validation set of our dataset (top three rows) and SEN12MS-CR-TS (bottom three rows). (Best viewed on screen zoomed-in)

of increasing size. We can see here that increasing the model size leads to higher reconstruction quality. Nevertheless, as this model also operates to much larger inputs, and considering that the complexity is $\mathcal{O}(n\sqrt{n})$, increasing the model size leads to significant increase in memory footprint. Due to this, 512 is the largest model size we could afford.

Model Size	Params	Validation Loss
64	0.5M	0.266
128	1.3M	0.251
256	3.9M	0.246
512	12.7M	0.241

Table 3. Ablation on upsampler model size.

We also consider the effect of the number of DOYs T forming the input tensor and the percentage of cloud coverage considered of the masks employed during training. The respective results are shown in Figure 2. We observe that increasing the number of DOYs from 3 to 5 leads to reduced validation loss. As the temporal horizon increases, the validation loss begins to be negatively affected. This can be attributed to changes happening in the captured area. Regarding cloud coverage, we observe that when masks with higher coverage are used during training, the validation loss decreases. This is reasonable as the model becomes more efficient in reconstructing larger areas affected by clouds. In any case, the differences in validation loss are small, showing that the proposed method is quite robust both with respect to the number of DOYs and the amount of mask cloud coverage employed during training.

4.4 Quantitative and qualitative results

Based on the ablation study, we consider in this section a core model of size 128 reconstructing the reduced resolution target image (64×64), followed by an upsampler model of size 512, that together produce the cloud-less target image in full resolution. We perform quantitative evaluation considering the Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) metrics with respect to the validation set of each dataset (Wang et al., 2004).

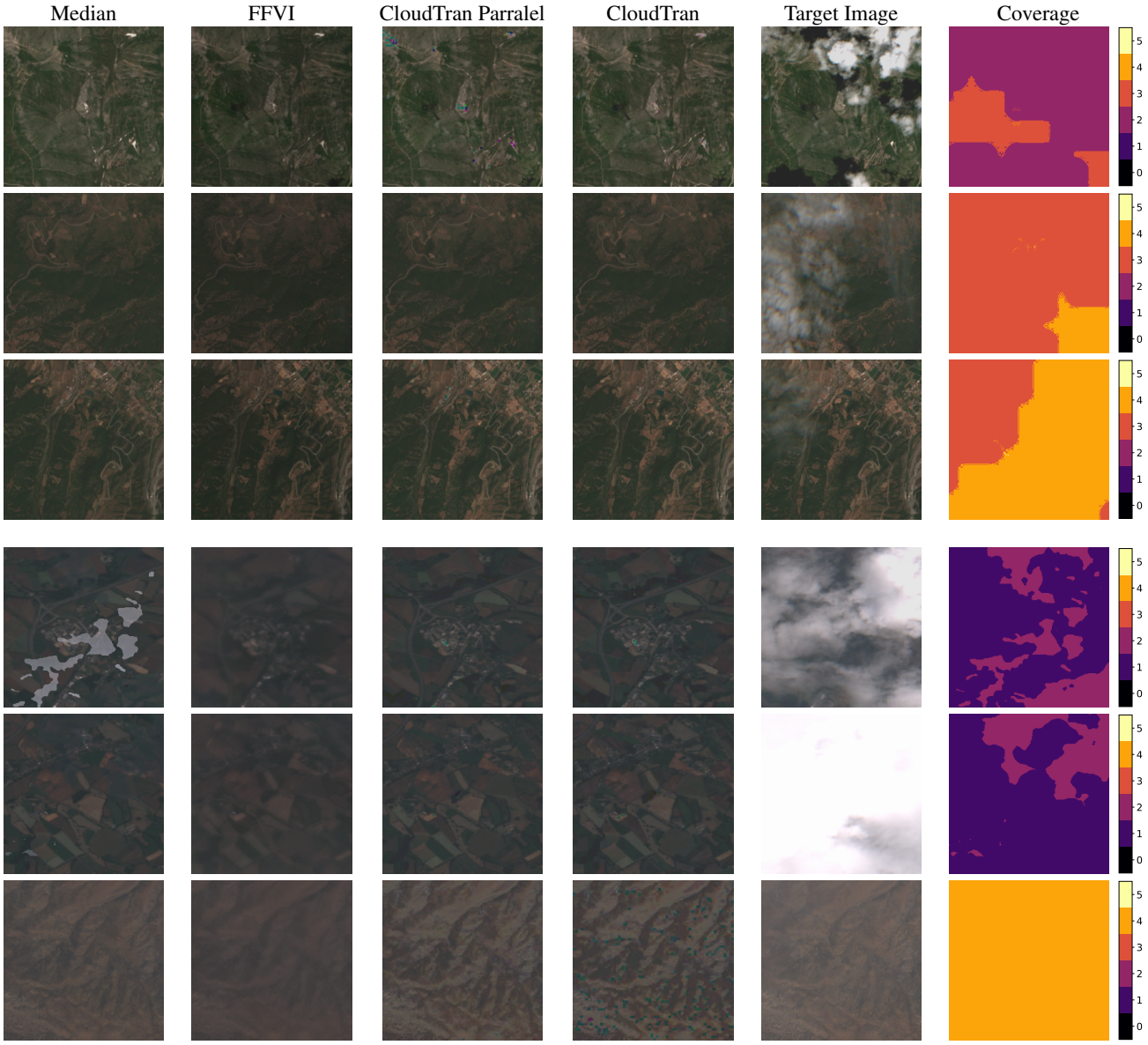

Figure 4. Random samples from test set of our dataset (top three rows) and SEN12MS-CR-TS (bottom three rows). (Best viewed on screen zoomed-in)

Tables 4 and 5 compare the performance of the proposed CloudTran method with different baselines for the low and full resolution outputs, respectively. In particular, we consider a suitably adapted version of the powerful video inpainting method of (Chang et al., 2019), as well as a standard gap-filling method based on rolling median filtering. Table 4, considers only the core model, while Table 5 report the results using the entire architecture. We also report the performance of the method proposed by (Pan, 2020) based on SpA GAN, as an indicative case of single-image cloud removal methods. SpA GAN does not use multiple dates, this translates to high MSE values. Our models outperform the baselines with a significant margin both for reduced and full resolution results, while using fewer trainable parameters than the second best FFVI.

Both tables also report the results obtained using solely the encoder output as CloudTran Parallel. Being a parallel model, its sampling is much more efficient, but the quality of the produced is inferior as they are affected more by artifacts. Nevertheless, they still perform better than other baselines.

We also report our results on the SEN12MS-CR-TS dataset. Our models have been trained on the SEN12MS-CR-TS data-

Method	**PSNR**(↑)	**SSIM**(↑)	**MSE**(↓)	**# Params**
SpA GAN (Pan, 2020)	33.03	0.9211	53.84	2.98M
Median	32.89	0.8540	55.720	NA
FFVI (Chang et al., 2019)	44.61	0.9796	2.977	35.9M
CloudTran Parallel (ours)	50.58	0.9891	3.098	3.2M
CloudTran (ours)	**54.09**	**0.9943**	**0.970**	

Table 4. Comparison with cloud removal baselines for 64×64 outputs.

Method	**PSNR**(↑)	**SSIM**(↑)	**MSE**(↓)	**# Params**
SpA GAN (Pan, 2020)	30.88	0.8916	78.98	2.98M
Median	32.46	0.8741	58.912	NA
FFVI (Chang et al., 2019)	48.31	0.9922	1.373	35.9M
CloudTran Parallel (ours)	50.26	0.9935	2.024	12.7M
CloudTran (ours)	**51.34**	**0.9950**	**1.202**	

Table 5. Comparison with cloud removal baselines for 256×256 outputs.

set for 100000 iterations. On the validation set, our full model achieves a PSNR of 50.00 dB, an SSIM value of 0.9931 and MSE equal to 6.426. These values are not directly comparable to the ones reported in (Ebel et al., 2022) for numerous reasons. Just to mention some of them, (Ebel et al., 2022) consider also data from Sentinel 1, processing is performed in a sequence to sequence fashion, and also validation is defined differently. Nevertheless, the values reported here on SEN12MS-CR-TS dataset are significantly improved with respect to the ones reported in (Ebel et al., 2022).

Figure 3 shows randomly chosen cloud removal results obtained by our models on the validation set of the two datasets considered. The first column shows the downsampled ground truth target image, the second shows the target image after applying the cloud mask, and the third one shows the cloud-free image produced by the core model. The next two columns show the output of the upsampler and the ground truth image in the original resolution. The last column shows for each pixel the number of valid pixels (cloud-free) in the entire input tensor.

Figure 4 shows randomly chosen results from the test sets of the two datasets considered. Here, the first two columns show the results of Median filtering and FFVI, respectively. The third and fourth columns show the results of the proposed CloudTran model from the encoder and the decoder, respectively. The original unmasked image and the cloud coverage are shown in the last two columns. We observe that the results of the proposed method contain fewer artifacts and are more faithful to the corresponding inputs.

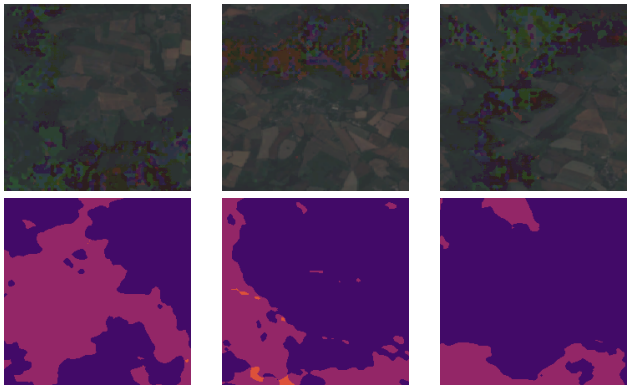

Figure 5. Failure cases (top row) and corresponidng cloud coverage (bottom row).

Finally, Figure 5 presents some failure cases. Our analysis suggests that these correspond to patches affected by severe ($> 50\%$) cloud coverage in all but 1 or 2 patches forming the input tensor. Increasing the cloud coverage percentage used during training, on the one hand, and the number of DOYs, on the other, can alleviate these effects.

5. CONCLUSIONS

Our work introduces a cloud-removal architecture based on two transformer-based models using axial-attention blocks for increased efficiency. An encoder-decoder model is proposed for producing low-resolution cloud-free images in an autoregressive fashion, given a number of input patches where the cloudy regions have been masked-out. An encoder-only parallel model is proposed for upsampling the cloud-free image to the original resolution. The proposed model is shown to perform significantly better with respect to a number of strong baselines, across an in-house and a large multitemporal sentinel-2 dataset.

ACKNOWLEDGEMENTS

This research is co-financed by Greece and the European Union (European Social Fund-ESF) through the Operational Programme "Human Resources Development, Education and Lifelong Learning" in the context of the project "Reinforcement of Postdoctoral Researchers - 2nd Cycle" (MIS-5033021), implemented by the State Scholarships Foundation (IKY).
We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPUs used for this research.

REFERENCES

Bazi, Y., Bashmal, L., Rahhal, M. M. A., Dayil, R. A., Ajlan, N. A., 2021. Vision transformers for remote sensing image classification. *Remote Sensing*, 13(3), 516.

Chang, Y.-L., Liu, Z. Y., Lee, K.-Y., Hsu, W., 2019. Free-form video inpainting with 3D gated convolution and temporal PatchGAN. *IEEE International Conference on Computer Vision*, 9066–9075.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S. et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Ebel, P., Xu, Y., Schmitt, M., Zhu, X. X., 2022. SEN12MS-CR-TS: A Remote Sensing Data Set for Multi-modal Multi-temporal Cloud Removal. *IEEE Transactions on Geoscience and Remote Sensing*.

Franchetti, B., Ntouskos, V., Giuliani, P., Herman, T., Barnes, L., Pirri, F., 2019. Vision Based Modeling of Plants Phenotyping in Vertical Farming under Artificial Lighting. *Sensors*, 19(20).

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y. et al., 2020. A survey on visual transformer. *arXiv preprint arXiv:2012.12556*.

Ho, J., Kalchbrenner, N., Weissenborn, D., Salimans, T., 2019. Axial attention in multidimensional transformers. *arXiv preprint arXiv:1912.12180*.

Karakizi, C., Karantzalos, K., Vakalopoulou, M., Antoniou, G., 2018. Detailed land cover mapping from multitemporal landsat-8 data of different cloud cover. *Remote Sensing*, 10(8), 1214.

Kumar, M., Weissenborn, D., Kalchbrenner, N., 2021. Colorization transformer. *International Conference on Learning Representations*.

Lin, C.-H., Tsai, P.-H., Lai, K.-H., Chen, J.-Y., 2012. Cloud removal from multitemporal satellite images using information cloning. *IEEE Transactions on Geoscience and Remote Sensing*, 51(1), 232–241.

Liu, L., Hu, S., 2021. SACTNet: Spatial Attention Context Transformation Network for Cloud Removal. *Wireless Communications and Mobile Computing*, 2021.

Liu, R., Deng, H., Huang, Y., Shi, X., Lu, L., Sun, W., Wang, X., Dai, J., Li, H., 2021. Fuseformer: Fusing fine-grained information in transformers for video inpainting. *IEEE International Conference on Computer Vision*, 14040–14049.

Liu, S., Marinelli, D., Bruzzone, L., Bovolo, F., 2019. A review of change detection in multitemporal hyperspectral images: Current techniques, applications, and challenges. *IEEE Geoscience and Remote Sensing Magazine*, 7(2), 140–158.

Menick, J., Kalchbrenner, N., 2019. Generating high fidelity images with subscale pixel networks and multidimensional upscaling. *International Conference on Learning Representations*.

Pan, H., 2020. Cloud Removal for Remote Sensing Imagery via Spatial Attention Generative Adversarial Network. *arXiv preprint arXiv:2009.13015*.

Papadomanolaki, M., Vakalopoulou, M., Karantzalos, K., 2021. A Deep Multitask Learning Framework Coupling Semantic Segmentation and Fully Convolutional LSTM Networks for Urban Change Detection. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9), 7651–7668.

Papamakarios, G., Pavlakou, T., Murray, I., 2017. Masked autoregressive flow for density estimation. *Advances in neural information processing systems*, 30.

Sebastianelli, A., Nowakowski, A., Puglisi, E., del Rosso, M. P., Mifdal, J., Pirri, F., Mathieu, P., Ullo, S. L., 2020. Sentinel-1 and Sentinel-2 Spatio-Temporal Data Fusion for Clouds Removal. *arXiv preprint arXiv:2106.12226*.

Singh, P., Komodakis, N., 2018. Cloud-gan: Cloud removal for sentinel-2 imagery using a cyclic consistent generative adversarial networks. *IEEE International Geoscience and Remote Sensing Symposium*, IEEE, 1772–1775.

Skakun, S., Wevers, J., Brockmann, C., Doxani, G., Aleksandrov, M., Batič, M., Frantz, D., Gascon, F., Gómez-Chova, L., Hagolle, O., López-Puigdollers, D., Louis, J., Lubej, M., Mateo-García, G., Osman, J., Peressutti, D., Pflug, B., Puc, J., Richter, R., Roger, J.-C., Scaramuzza, P., Vermote, E., Vesel, N., Zupanc, A., Žust, L., 2022. Cloud Mask Intercomparison eXercise (CMIX): An evaluation of cloud masking algorithms for Landsat 8 and Sentinel-2. *Remote Sensing of Environment*, 274, 112990.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. *Advances in neural information processing systems*, 5998–6008.

Wang, Z., Bovik, A. C., Sheikh, H. R., Simoncelli, E. P., 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612.