Unsupervised Image-to-Image Translation with Shared Efficient Attention Mechanism

Jose C. Castelo, DoHoon Lee *

Pusan National University, Busan (South Korea)

ABSTRACT

Deep Learning have allowed for new approaches to modifying digital images. As a result, tasks like Image-to-Image (I2I) translation have emerged with the goal of transforming the aspect of images in a domain “X" to resemble the characteristic features of another domain “Y". E.g. the translation of day (“X") to night (“Y") and vice versa. Generative Adversarial Networks (GAN) in conjunction with a cycle consistency loss have become the most prevalent method to perform image translations. Nevertheless, the incorporation of self-attention mechanisms into GANs remains a challenge due to the memory-intensive nature of GANs and the quadratic complexity of this attention mechanism. Given the performance boost that attention modules have shown in different computer vision tasks, it is desired to find alternative ways to converge both techniques. To address this issue, this paper utilizes an efficient-attention mechanism to introduce a novel architecture denominated EAGAN for the I2I translation task. Furthermore, the architecture makes use of an attention-sharing technique to re-utilize the computed attention-maps in the reconstruction of the translated image. A series of experiments are performed to evaluate and compare the proposed architecture against competing methods. The experimental results showed improvements in the KID scores for object and scenery translation tasks exposing the benefits of the proposed architecture and its attention-sharing mechanism.

KEYWORDS

Attention Mechanism, Deep Learning, Generative Adversarial Networks, Generative Models, Image Translation.

I. Introduction

Image manipulation has been an active area of study since the early days of photography. Different methods have been developed to solve tasks like image composition and retouching. As described in [1], [2], some of the initial image manipulation methods required the direct manipulation of image negatives. However, the development of computers and the high availability of digital images have push the scientific interest in the development of algorithms and techniques that aim to modify the visual aspect of images.

More recently, the utilization of Deep Learning techniques, specifically Convolutional Neural Networks (CNN) as introduced in [3] have shown great capabilities when working with images for tasks like classification, segmentation, object detection, etc. Furthermore, convolutional architectures have also been utilized in methods that allow to modify the visual aspect of images as in [4] that transfers the artistic style of well-known artworks to photographs.

While generative models like Generative Adversarial Networks (GAN) [5] were initially designed to create novel samples from a given image distribution, these architectures have also been utilized in a conditional way to force the modification of images rather than the creation of novel samples of a given distribution. Works like [6] extensively explored this tasks known as Image-to-Image (I2I) Translation for translating samples from image domain “X" to contain the visual aspect of images from domain “Y". The utilization of these kind of methods requires the minimum training of a pair of neural networks (generator and discriminator), with some similar works like [7], [8] proposing the utilization of multiple discriminators, increasing the number of networks to train up to six. While effective, approaches like these can quickly reach expensive memory requirements that can become a challenge for further architecture modifications.

One of the desired modifications to these type of frameworks for I2I translation would be the introduction of an attention mechanism for the translation process. After popularized with the introduction of the transformer architecture in [9], the utilization of attention has shown a boost in performance for tasks like classification and object detection as demonstrated in [10]. However, even when GAN based methods have been highly adopted for image generation and I2I translation tasks, the utilization attention mechanisms have been limited to the solely introduction of attention blocks after significantly down-sampling the input image as explored in [11], [12].

Even when the mentioned works have obtained a performance improvement, it is still desired to include attention layer through the entire architecture. For the I2I translation task, this can result more significant in locations closer to the stem or output layers where the image is closer to being a well-defined real or fake image allowing the learning of significant long-range dependencies to better assist the translation process. In an effort to address this situation, the present paper explores the utilization of alternative attention mechanisms not only to introduce attention blocks to a I2I translation architecture, but to re-utilize the long-range dependencies computed near the stem layer for the reconstruction layers closer to the output.

* Corresponding authors:
E-mail address: mail@mail.com (First A. Author), mail@mail.com (Second B. Author), mail@mail.com (Third C. Author).

In order to asses the effectiveness of the resulting architecture and the proposed attention sharing mechanism, experiments for the object and scenery translation sub-tasks were performed. The results of the proposed approach are compared with alternative attention-based methodologies for I2I translation. The comparison shows a general improvement in the KID evaluation metric for the majority of the revised tasks, more significantly, in the scenery translation tasks. In essence, the main contributions of this paper can be seen as follow:

- Construction of an architecture for the I2I translation task with integrated efficient attention mechanism.
- Proposed of an attention-sharing mechanism to take advantage of the computed long-range dependencies to improve the translation process.
- Conduction of experiments to compare the performance of the proposed method against different alternative methodologies.

In the remaining sections of this paper an overview of the related works is presented in Section II. Section III describes the proposed architecture and attention-sharing mechanism. The experimental results are presented in Section IV and further discussed in Section V. Finally, a conclusion is presented in Section VI.

II. Related Work

A. Generative Models

Convolutional Neural Networks have prove their capabilities to work with digital images in tasks like classification, segmentation and object detection. Subsequently, the introduction of Generative Adversarial Networks opened the door for utilizing similar CNN for an image generation task. A general GAN framework consists of a pair of networks which are presented with opposing tasks. On one side, a generator G attempts to generate novel samples that resembles the training data distribution. On the other hand, a Discriminator network D is trained to accurately estimate the probability of an input belonging to the real or generated distribution by G. This is achieved by having G and D playing a mini-max game on the adversarial loss function in Equation 1.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))] \quad (1)$$

Initial implementations of GANs were not capable of controlling the class of the produced images. E.g. when training a GAN with the MNIST [13] dataset that presents samples of handwritten numbers from 0 to 9, it wasn't possible to specify which number to generate. Conditional GANs (cGAN) [14] addressed this by providing the generator and discriminator with additional information regarding the target class. Similarly, further works have take a similar approach to condition GANs on an image input. More notable, [6] proposed a general purpose framework for the I2I translation task. However, its utilization was limited for working with a set of paired images, meaning the ground truth images for each of the image domains involved in the translation.

To allow I2I translation on unpaired data, Zhu et al. [15] proposed a GAN architecture widely known as CycleGAN. This architecture is based in the so called "cycle consistentcy" idea. Similar to how when translating a sentence from language A to language B, then translating it back from B to A should get us back the original sentence, The image translation should present the same consistency. To achieve this two generators and discriminator are utilized to asses each of the translation directions ($A \rightarrow B$ and $B \rightarrow A$). To enforce the cycle consistency a new term is introduced to the objective function where the $L1$ distance between the original input and backward translated outputs is minimized as follows:

$$\mathcal{L}_{cyc}(G,F) = \mathbb{E}_{x\sim p_{data}(x)}[\|F(G(x))-x\|_1] + \mathbb{E}_{y\sim p_{data}(y)}[\|G(F(y))-y\|_1] \quad (2)$$

The CycleGAN framework has been extensively explored in a wide variety of tasks like attribute-guided face generation [16] and music genre transfer [17]. Additionally, a popular application for I2I translation methods is the generation of novel data samples as an augmentation in the training of models for tasks like COVID-19 detection [18] and wildfire detection [19].

Beyond these applications, the CycleGAN framework has served as a base for different works that aim to improve the capabilities of I2I translation. For instance, Attention-guided Generative Adversarial Networks (AGGAN) [20] introduce a forking network after the generator's bottleneck to learn different attention masks that focus on the most discriminate features for a given translation task. In this way, the model can focus on changing the most import features for a successful image translation while keeping unimportant areas like the background unmodified.

Another example is Attention-Based Spatial Guidance for Image-to-Image Translation (ASGIT) method [21]. This method utilizes the fourth layer of the discriminator to generate an attention map based in the most discriminative features of each translated sample in the dataset. These attention maps are then utilized to guide the generator on which features of the input required the most improvement to fool the discriminator. In this sense, this method utilize an attention mask to guide the generator into improving the translation process.

Similar to these methods, the present work utilize the CycleGAN framework as a base to develop a new architecture that utilizes an alternative attention mechanism in an effort to improve the image translation process.

B. Attention Mechanisms

While convolutions can efficiently model dependencies between pixels inside the kernel neighborhood, long-range dependencies could only modeled by utilizing several layers. Even then, dependencies could be overlooked if the model is not deep enough or if a vanishing gradient problem arises. Attention mechanisms surge as a method to model long-range dependencies that result essential in sequence-to-sequence (seq2seq) models utilized for problems like Natural Language Processing (NLP). [22], [23] introduced the bases for the so-called "self-attention" mechanism which was later used as the main component of the transformer architecture [9] that have pushed the NLP field and have been subsequently adapted to perform computer vision tasks with promising performance on classification and object detection [24], [25].

The main inconvenience of self-attention modules is the quadratic complexity and high memory requirements. This becomes a greater problem when complex architectures or high quality inputs are utilized. To overcome this

limitation, works like [26], [27] have proposed less resource-intensive alternatives to capture long-range dependencies. Nevertheless, this methods produce an approximation by utilizing lower-rank matrices or present a limitation to encode information given its usage of scalars after flattening the input vector. Alternatively, Efficient Attention mechanism [28] takes advantage of the associative property of matrix multiplication to propose a mathematically equivalent mechanism to self-attention. Differently to self-attention, this method interprets each channel of the keys as a global attention map that does not correspond to a specific position but to a semantic aspect of the input image.

The reduced computational resources of the efficient attention mechanism allows for its wide utilization in generative models like GANs where self-attention have been limited to single layers in a low dimensional space. Other methods makes use of a scaling attention variant used to emphasize import features for the process. This paper makes use of the efficient-attention module and its unique interpretation of the keys channels as attention-maps to implement an attention-based architecture with an attention-sharing mechanism for the I2I translation task.

III. Efficient Attention GAN

A. Overview

As a general I2I translation problem, we consider two image domains X and Y with data instances $x \in X$ that follow the distribution P_x and $y \in Y$ follows distribution P_y. The objective of this work is to learn two mapping function represented by the generators $G : x \rightarrow y$ and $F : y \rightarrow x$ such that the distributions $G(X)$ and $F(Y)$ are indistinguishable from Y and X respectively.

The approach introduced in this paper paper incorporates efficient-attention modules to both generator networks to construct the denominated "Efficient Attention Generative Adversarial Network" (EAGAN). The objective of this architecture is not only to integrate the attention mechanism to the translation process, but to re-utilize the attention maps computed from the source domain X when reconstructing a translated sample from the target domain Y.

As the Pix2Pix [6] paper states, in an I2I translation tasks a big amount of low-level features are shared between the input and output images. A similar argument can be made for the long-range dependencies computed by attention mechanisms. If a source image and the ideally translated image are expected to be similar in structure, the attention maps in the source domain and the target domain should be relatively similar. E.g. the attention given to pixels of an object like a car should be the same in an original day scene and a translated night scene. In this way, the sharing of the attention maps from the source domain to the reconstruction of the target domain leads to a better consistency in the aspect of the elements contained in the translated scene.

B. Architecture

The Generator and Discriminator architectures introduced by CycleGan [15] have shown great robustness for the execution of this task. This has led to various I2I translation works utilizing such architecture as a baseline for developing new state-of-the-art methods. This work makes use of the CycleGAN generator as a foundation to explore the incorporation of the efficient-attention module and the attention-sharing mechanism.

This baseline generator consist of three stages that, intuitively, perform different sub-tasks for the image translation process.

- Encoder: extracts meaningful features from the input image. The image is converted to a latent feature space via a series of convolutions that down-samples the dimensionality of the input.
- Bottleneck: employs a series of residual blocks that learn to change the extracted features from the source domain to the target domain. In this stage, the dimensionality of the features remains unchanged. The number of residual blocks can be treated as a hyper-parameter to balance between speed and quality of results.
- Decoder: reconstructs a translated sample of the same resolution as the input. For this, a series of transposed convolutions are utilized to upscale the features to the desired dimension and a final convolution layer maps the output to a three-channel RGB image.

Given that it is desired to learn long-range dependencies in the source domain, then re-utilize these dependencies while reconstructing the target domain sample, the ideal location to introduce the attention blocks are the encoder and decoder sections in the generator network. In this stages, the features closely correspond to a well formed image in either of the domains, in this way, the computed attention maps can be shared through the mirrored attention modules on layers i and $i - L$ being L the number of layers in the network.

An additional consideration needs to be taken in the positioning of the efficient-attention modules. A first option consist on placing the attention layers after the down-sample convolution in the encoder and before the transposed-convolution in the decoder. Alternatively, a second option can consist on the placement of the attention modules prior to the convolution in the decoder and after the transposed-convolutions in the decoder.

To determine the better location for the attention modules, experiments to compare the performance of both options were conducted in a small scale utilizing a scenery translation datasets. Following the results presented in Table 1, the second option lead to an improved performance on the majority of the tested tasks. Accordingly, this placement option is adopted in the rest of the present work.

Given that it is easier for the network to learn dependencies in a local neighborhood, in the initial training steps it is desired to rely more on these local dependencies and wait for the long-range dependencies to become more meaningful before considering them more heavily. To achieve this, similarly to [29], a learnable scale parameter is introduced to multiply the output of the attention and subsequently add the input features back. In this fashion, the final output of the introduced efficient attention modules is given by,

$$y_i = \gamma o_i + x_i \tag{3}$$

where γ is initialized to zero to initially focus on the easier task of producing good features based on the local neighborhood with the convolution then assigning a higher weight to the long-range dependencies learned from more meaningful features.

Option	Night↔Day		Snow↔No Snow		Rain↔No Rain		Fog↔No Fog	
	N→D	D→N	S→NS	NS→S	R→NR	NR→R	F→NF	NF→F
1st	**1.81**	**1.23**	3.98	2.56	3.69	**1.45**	4.41	4.67
2nd	1.94	1.34	**3.60**	**2.55**	**3.38**	1.56	**4.15**	**4.44**

Table 1: Performance results from the architecture with efficient attention module placed before (1st) and after (2nd) the convolution blocks. Values are presented in KID terms and scaled by 100. Lower value is better.

C. Attention Sharing

Different to dot product self-attention, efficient attention interprets each channel of the keys $K \in \mathbb{R}^{n \times d_k}$ as a single attention maps that, instead of corresponding to the attention between two points in an image it learns to represent a semantic aspect of the images. Since the input and output images are expected to be highly related in their content, it can be inferred that the attention between features in the early layer i of the generator architecture can be reused on a corresponding ending layer $i - L$. Given that the original attention is computed in real samples, sharing these attention maps becomes beneficial when reconstructing the translated image to produce more consistent and realistic outputs. Following this, a series of skip connections are placed between each of the Key vectors of the mirrored efficient attention blocks to share the long-range dependencies between both ends of the architecture. Although the attention maps from the Key network are shared between a pair of attention blocks, the block located in the decoder maintains independent Query and Value trainable networks.

A refinement network is placed between each of the skip connections to refine the attention maps coming from the original domain. Considering that the efficient attention mechanism interprets each channel as an individual attention map, this refinement network is implemented as a depth-wise convolution with a 3×3 kernel. This special type of convolution divides the input channels into a given number of groups and utilizes a different kernel for each of these groups. For this specific case, our architecture utilized d_k groups to ensure each of the channels from Keys is convolved by a corresponding kernel. In this way, no mixing among the different attention maps is ensured during the refinement. This attention-sharing mechanism is illustrated in Figure 1.

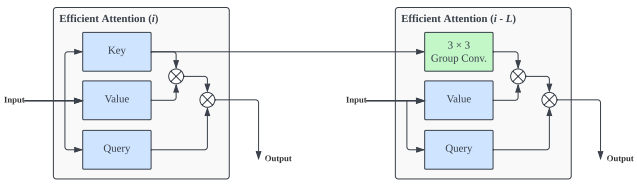

Fig. 1. Attention Sharing Mechanism for EAGAN. The Key, Value, and Query are implemented as convolution operations. The efficient attention mechanism interprets the Key result as an attention maps which are shared between corresponding i and $i - L$ layers. The attention maps are passed through a 3×3 grouped convolution to refine the attention maps while ensuring no information mixing among the different channels.

Analogous to how the number of channels grows through the layers of a deep convolutional neural network, the number of attention maps d_k can vary across the architecture and be tuned as a hyper-parameter. For this implementation, the number of attention maps d_k is set to half the number of input channels on each layer. However, considering the importance of the initial attention block, which learns dependencies directly from the original 3-channel input, the number of attention maps to compute for this initial block is fixed to a value of eight for the d_k dimension. Figure 2 presents the described EAGAN architecture.

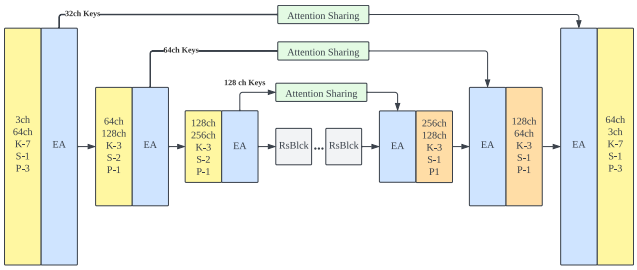

Fig. 2. Generator architecture for the proposed EAGAN. This architecture introduces efficient-attention modules on the encoder and the decoder in the position that obtained the best performance as shown in 1. The introduced attention modules make use of the attention-sharing mechanism shown in 1.

D. Objective Function

Following the standard training of a GAN network, the discriminators and generators are trained simultaneously by playing a min-max game on an adversarial objective function. Rather than the standard adversarial loss, the alternative least-square error loss proposed by [30] was utilized for the training due to its improved stability and higher quality results. In this sense, the objective function for the discriminator and generator becomes as presented in Equation 4.

$$\begin{aligned} \min_G \mathcal{L}_{GAN}(G) &= \mathbb{E}_{x \sim p_{data}(x)}[(D(G(x)) - 1)^2] \\ \min_D \mathcal{L}_{GAN}(D) &= \mathbb{E}_{y \sim p_{data}(y)}[(D(y) - 1)^2] + \\ & \quad \mathbb{E}_{x \sim p_{data}(x)}[D(G(x))^2] \end{aligned} \tag{4}$$

The proposed framework learns the translation mappings by enforcing a cycle consistency between the generator. This is, after passing an image through both generators, the image should be as close to the original as possible given that the translation should be reverted. Therefore, additionally to the adversarial loss, a cycle consistency term shown in Equation 2 is set to be minimized.

Given the importance of the colorization aspect for the scene translation task, it is desired to preserve the characteristic colors that are commonly present in a given image domain. For this, an identity loss term is introduced to the objective function as proposed by [31]. This term encourages the generators to avoid unnecessary changes to the input when this is already a sample from the target domain. Additionally, this encourages the generator to learn and preserve the tint and coloration more closely. The identity term of the loss function is given as follows:

$$\mathcal{L}_{\text{identity}}(G,F) = \mathbb{E}_{y\sim p_{data}(y)}[\|G(y)-y\|_1] + \mathbb{E}_{x\sim p_{data}(x)}[\|F(x)-x\|_1] \quad (5)$$

The Full objective function for the training of the models is composed of the adversarial term, cycle consistency term, and the identity loss. The resulting loss function utilized in the training process is given as follows:

$$\begin{aligned}\mathcal{L}(G,F,D_X,D_Y) = {} & \mathcal{L}_{\text{GAN}}(G,D_Y,X,Y) + \\ & \mathcal{L}_{\text{GAN}}(F,D_X,Y,X) + \\ & \mathcal{L}_{\text{cyc}}(G,F) + \\ & \mathcal{L}_{\text{identity}}(G,F) +\end{aligned} \quad (6)$$

IV. Experiments and Results

In order to evaluate and compare the performance between the proposed architecture against competing solutions, a series of models were trained from scratch on data sets for object and scenery translation sub-tasks. This section describes the experiment condition in detail and presents the results of such experiments.

A. Data sets

While the proposed architecture presents a general framework that can learn translation functions in a wide variety of domains, it is desired to observe the performance on slightly different image translation tasks. In certain I2I translation problems the area of the image that defines a successful translation is located in a reduced area. This situation often occurs on the denominated object translation task, where the area objective is to translate a determined object on an image. On the other hand, a more complex case requires the translation of the entire scene to obtain a successful translation; this task is known as scenery translation. Data sets that present both of these scenarios are utilized in the experiments to evaluate the performance of the proposed solutions.

1. Object Translation

Object translation is a sub-task where the objective is to translate a specific kind of object that is presented in an image. The environment (image background) where such objects are located varies through the image samples and since it does not contribute to the quality of the translation, it is often desired to maintain this information unchanged.

For the consideration of the Object Translation sub-task, two popular data sets for I2I translation were selected. The apple2orange data set [15] consists of unpaired image samples of orange and apple images that vary in location, quantity, aspect, and environment. A second data set focused on the object translation sub-task is the well-known Horse2Zebra [15]. Similarly, it presents unpaired images of horses and zebras in a variety of conditions. Samples from each class of these data sets are displayed in Figure 3.

2. Scenery Translation Data

Scenery Translation is presented when the areas of interest to translate are spread all across the image rather than in a contained area. This requires the model to learn a domain translation for each of the elements that can be presented in a given image. In this way, scenery translation is often a more difficult task than object translation.

Fig. 3. Samples from the different classes present on the dataset for the object translation task. These samples show a variety of conditions in size, quantity, orientation, and number.

For the scene translation task the "Adverse Conditions Data set with Correspondences" (ACDC) [32] data set is utilized. This data set presents images of driving scenes in different environmental conditions: night, snow, rain, and fog. For each adverse condition image, a corresponding scene taken under normal conditions is available. Even though the data set provides an approximation of paired samples, the data set is utilized in an unpaired style.Samples of the different image domains can be seen in Figure 4.

Fig. 4. Samples from the different meteorological conditions presented in the ACDC dataset. *The day condition presents a combination of samples from the optimal corresponding images from the different adverse conditions.

B. Experiment Details

A model is trained from scratch on each pair of image domains from the data sets. The models are trained for 200 epochs each, where the learning rate for both, the generator and discriminators, is set to 0.0002 during the first 100 epochs and linearly decays to zero in the subsequent 100 epochs. The weights of the generators and discriminators were initialized from a Normal distribution $\mathcal{N}$ of mean

$\mu = 0$ and standard deviation $\sigma = 0.02$. A batch size of one was utilized in conjuction with an Adam optimizer [33] with $\beta_1 = 0.5$ and $\beta_2 = 0.999$ for both, the discriminator and generator networks.

Following the original implementation, we set the importance of the cycle consistency loss utilizing a $\lambda\mathcal{L}_{cyc} = 10$. Additionally, the models were trained with the identity loss to avoid color shifting in the translations as reported in [15]; The regularization term for the identity loss is set as $\lambda\mathcal{L}_{\text{identity}} = \lambda\mathcal{L}_{cyc} * 0.5 = 5$.

During the training, each of the images is resized to a 256×256 resolution using the bi-linear interpolation method. The resulting image is then individually normalized by subtracting the mean and dividing by the standard deviations of the entire data set. The training for all the different models was performed in a machine running the Linux distribution Pop!_OS 21.10 with a single graphic card Nvidia GeForce RTX 3090.

C. Evaluation Method

While a variety of methods like [34], [35] have been proposed to assess the quality of Deep Learning generated images, this is still an open problem where no consensus has been met by the research community. Nevertheless, the KID [36] metric have seen a wide adoption on recent literature. This evaluation method makes use of a pre-trained InceptionV3 [37] model to obtain a low-level representation of the features in the real and generated distribution allowing to measure the similarity these two. This paper makes use of the KID score as the main comparison metric. More specifically, the torch-fidelity package [38] is utilized as it makes use of the original weights from the InceptionV3 model, ensuring accurate results with a negligible difference.

The training of GAN architectures is characterized by its sensitivity to encounter problems like mode collapse or sudden divergence. For this reason, it is a common approach to constantly evaluate the models during the training process to select the best performing model. We evaluate the models after each epoch of the training process and save the current checkpoint if there if a performance improvement is present.

D. Experimental Results

For the comparison between the different architectures, the lowest KID score obtained by each model during its training is selected as a reference of its performance. Table 2 and 3 presents a comparison between the different architectures in terms of KID scores for the scenery and object translation tasks respectively.

When observing the results of the scenery translation task on table 2, a general performance improvement can be observed on most of the translation tasks with the exception of the night-related tasks where the simpler CycleGAN model performs the best closely followed by the proposed architecture. Overall, the proposed EAGAN with the cross-domain attention-sharing mechanism showed superiority in the translation of different adverse condition scenes with the KID metric as reference.

On the other hand, the experiments on object translation datasets also showed improvements in performance by the proposed architecture. The EAGAN architecture emerged as the best-performing model for the "apple to orange" and "horse to zebra" tasks while remaining fairly close to CycleGAN for the "orange to apple" task.

When working with images, a visual comparison is a straightforward approach to evaluate different GAN architectures. However, human-eye evaluations will always present a preference bias making it unreliable. Therefore, evaluation metrics like KID are preferred, however, a visual inspection is still interesting to visualize the produced outputs by the different models. For this, Figures 5 and 6 present the resulting outputs from the different trained for the scenery and object translation tasks respectively. The displayed images are produced based on the same input image by the model checkpoints with the lowest KID score during the training process.

Fig. 5. Visual comparison of the output from the different trained models for the scenery translation task when the same input image is given.

V. Discussion

A. Detailed Visual Inspection

While visual inspection of generative models is not an ideal way of evaluation, the examination of small details present on the generated images can help to bring a better understanding and interpretability of the quantitative scores by identifying areas where some models are superior to others.

Figure 7 presents an example of the “Horse to Zebra" tasks where an area of interest is zoomed-in for a more detailed inspection. While all the generated images present the characteristic black and white stripes of a zebra, it can

Method	Night↔Day		Snow↔No Snow		Rain↔No Rain		Fog↔No Fog	
	N→D	D→N	S→NS	NS→S	R→NR	NR→R	F→NF	NF→F
CycleGAN [15]	**1.09**	**0.64**	1.75	2.23	2.53	1.05	1.75	2.23
AGGAN [20]	6.84	7.27	1.94	2.16	2.73	1.36	1.94	2.16
ASGIT [21]	5.22	2.47	1.89	2.87	3.20	1.40	1.89	2.87
EAGAN	1.23	0.83	**1.72**	**1.62**	**1.99**	**1.02**	**1.72**	**1.62**

Table 2: Performance comparison for the scenery translation task between the proposed architecture and alternative methods. KID values scaled by 100 where a lower value indicates better performance.

Method	Orange ↔ Apple		Horse ↔ Zebra	
	O → A	A → O	H → Z	Z → H
CycleGAN [15]	2.83	6.63	3.11	0.88
AGGAN [20]	9.55	10.26	3.25	**0.66**
ASGIT [21]	5.56	7.98	3.35	1.30
EAGAN	3.00	**6.28**	**2.75**	0.98

Table 3: Performance comparison for the object translation task between the proposed architecture and alternative methods. KID values scaled by 100 where a lower value indicates better performance.

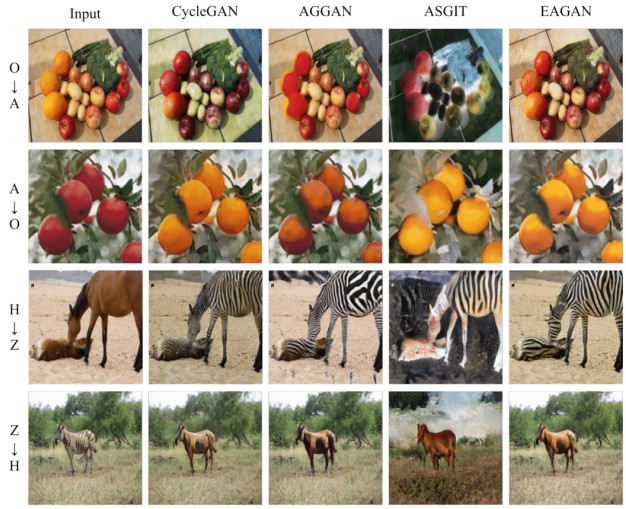

Fig. 6. Visual comparison of the output from the different trained models for the object translation task when the same input image is given.

be argued that some models do a better job at this task. CycleGAN produces a translation that still presents patches where the brow color of the original image is still present. AGGAN produces stripes that can result too thin to be considered realistic; additionally, it translates the area of the tail which should not have the stripes pattern. While the ASGIT model shows some of the required characteristics of a zebra, the color shifting of the output is evident. Finally, EAGAN presents a consistent stripes pattern with some minor artifacts in the tail section.

Figure 8 presents an example of the scenery translation sub-task, specifically for the "Snow to No-Snow" case. This sample image presents a challenging situation where a pedestrian is present in the image occluding a section of the street that contains snow. In this situation, CycleGAN and AGGAN show difficulties preserving the feet of the subject while the trunk area tends to blend with the surroundings. ASGIT seems to conserve the feet better while having a more distinguishable coloring in the trunk; However, it presents a more noisy and grainy aspect in the whole image. EAGAN shows better preservation of the feet and

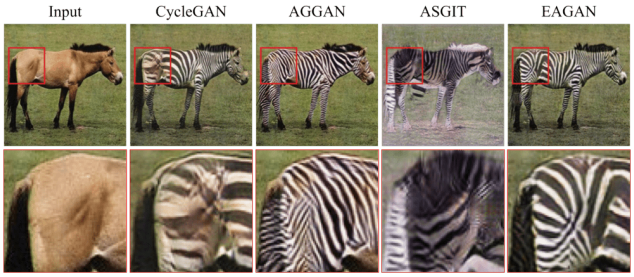

Fig. 7. Detail inspection of the output from the different models for the object translation task. The images in the second row present a zoomed-in window of the red square in the image above.

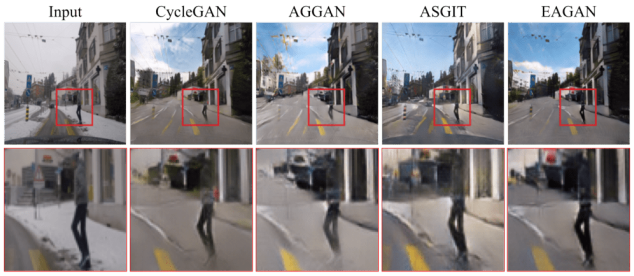

Fig. 8. Detail inspection of the output from the different models for the scenery translation task. The images in the second row present a zoomed-in window of the red square in the image above.

trunk area while maintaining a less noisy look

As already stated, a visual evaluation is not an optimal method to evaluate translated images. While multiple human evaluators can agree on certain aspects, other ones can depend on personal preference introducing a bias and inconsistency. Furthermore, it can result in impossible or very expensive to evaluate every single sample contained in the evaluation set. For this reason, even when a visual inspection is presented for the sake of completeness, it is always preferred to refer to metrics like KID as utilized in the results Section D.

B. Attention-Map Visualization

An interesting aspect to observe from the trained EAGAN model is the aspect of the computed attention maps from a given input image. In this step, it is expected that the efficient attention modules produced attention maps that focus on different areas of an image. For the visualization of such attention maps the initial attention module is selected given the matching dimension of the height and width with the original input images making it easier to interpret the attention maps. Each of the channels from the Key vector in the attention module is passed through the Soft-max function and expanded to a range of

0 and 1 by subtracting the minimum value and dividing by the maximum value for each attention map. With this, the attention maps can be visualized as gray-scale images where the brighter pixels indicate higher attention on that area of the input.

The extracted attention maps are presented in the intermediate columns of Figure 9. Here, it is possible to observe how each attention map can attend to different areas of the image. This behavior is more noticeable in the object translation task. E.g. the second row presents an apple to orange translation task where the first attention map seems to attend to the apple, the second one to the bushes, and the final map attends to the sky. The scenery translations present a broader task where the entire image requires to be translated, for this reason, this behavior results less evident, however, it is still possible to see how the attention is distributed to different areas of the image.

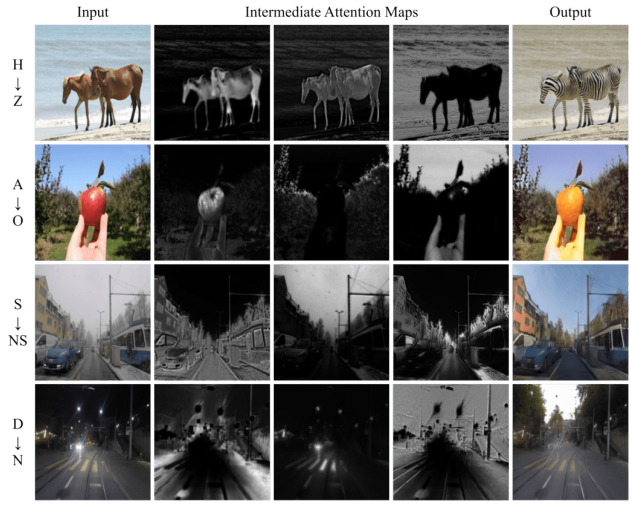

Fig. 9. Visual inspection of some of the attention maps produces by the initial attention module in the EAGAN model. The first and last columns show the input and the translated output while the intermediate images correspond to some of the attention maps.

C. Computational Requirements

The computational requirements of the different models are another area that can be discussed. As one of the motivations for this work was the difficulty of incorporating the Self-Attention mechanism into the I2I translation process due to its intensive memory requirements, the requirements for the different models and the Self-Attention mechanism are presented in 4.

Considering that the proposed EAGAN introduce additional layers to the base CycleGAN generator, an increment in the required FLOP and parameters is expected. Nevertheless, the proposed EAGAN architecture manages to keep lower requirements than alternative methods like AGGAN while showing better performance. For the memory requirements, the increased amount of megabytes is due in part to the need of maintaining some feature maps computed in the encoder to re-utilize them in the decoder section.

As the ASGIT method utilizes the same generator architecture as a standard CycleGAN, it presents the same number of FLOP, parameters, and memory requirements. However, it is important to consider that during the training, it requires to reserve memory for the attention mask of each sample present in the dataset. This characteristic can rapidly become a bottleneck for the training process when a considerably large dataset is utilized.

Method	FLOP(M)	Params.(M)	Mem.(MB)
CycleGAN [15]	49,616	11.383	470
AGGAN [20]	57,097	11.822	566
ASGIT [21]	49,616	11.383	470
SA Module*	208,861	0.005	21,660
EAGAN	53,273	11.688	782

Table 4: Comparison of the number of floating-point operations (FLOP), trainable parameters, and memory requirements for the generator network of the different models. FLOP and parameters are presented in millions (M) scale while memory is in megabytes (MB). FLOP and memory are measured on a feed-forward pass of the generator network set to inference mode. * represent a single Self-Attention module as implemented in [29] with a reduced input of $232 \times 232 \times 64$.

An interesting point arises when comparing the requirements for a single Self-attention module. The computations required for an input tensor of size $232 \times 232 \times 64$ (approximate input for the first attention module in EAGAN) dramatically surpass the FLOP and memory required by the complete architecture of all the different methods. This comparison illustrates how an architecture and an attention-sharing mechanism like proposed in this work would result inviable by utilizing the Self-Attention mechanism.

VI. Conclusion

The introduction of attention mechanisms to the Image-to-Image translation field has result a challenging task. The already memory-demanding process has resulted in a prohibitive utilization of the self-attention mechanism due to the also high memory requirements. The present paper introduce the EAGAN architecture as a mean to mitigate this limitation. This is done by utilizing the alternative efficient attention mechanism to integrate an attention mechanism through key layers of a GAN generator network.

Beyond the integration of the efficient attention module in the architecture, a key contribution is made with the proposed cross-domain attention-sharing mechanism introduced in the EAGAN architecture. This mechanism takes advantage of the interpretation given by the efficient attention module to allow the re-utilization of the long-range dependencies computed from the real images at the time of reconstructing its translated version. In this way, a new approach to obtain the benefits of attention mechanisms while maintaining the viability of its usage given its reduced memory footprint is introduced.

The conducted experiments and comparison with alternative methods for I2I translation showed a performance superiority by the proposed architecture. More specifically, the EAGAN model obtain the best performance in most of the evaluated tasks when utilizing KID as the evaluation metric demonstrating the benefits that the attention mechanisms bring to the image translation field and that has been held back due to the restrictive memory requirements of the processes.

References

[1] M. Hofer, K. O. Swan, "Digital image manipulation: A compelling means to engage students in discussion of point of view and perspective," *Contemporary Issues in Technology and Teacher Education*, vol. 5, no. 3, pp. 290–299, 2005.

[2] R. Pettersson, "Image manipulation," *Media and Education*, pp. 20–23, 2002.

[3] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[4] L. A. Gatys, A. S. Ecker, M. Bethge, "A neural algorithm of artistic style," *Journal of Vision*, vol. 16, p. 326, 8 2015, doi: 10.48550/arxiv.1508.06576.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, vol. 27, 2014, Curran Associates, Inc.

[6] P. Isola, J.-Y. Zhu, T. Zhou, A. A. Efros, "Image-to-image translation with conditional adversarial networks," *CVPR*, 2017.

[7] J. Zhao, J. Zhang, Z. Li, J.-N. Hwang, Y. Gao, Z. Fang, X. Jiang, B. Huang, "Dd-cyclegan: Unpaired image dehazing via double-discriminator cycle-consistent generative adversarial network," *Engineering Applications of Artificial Intelligence*, vol. 82, pp. 263–271, 2019, doi: https://doi.org/10.1016/j.engappai.2019.04.003.

[8] J. Park, D. K. Han, H. Ko, "Adaptive weighted multi-discriminator cyclegan for underwater image enhancement," *Journal of Marine Science and Engineering*, vol. 7, no. 7, 2019, doi: 10.3390/jmse7070200.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, Curran Associates, Inc.

[10] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, J. Shlens, "Stand-alone self-attention in vision models," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, Curran Associates, Inc.

[11] L. Wang, L. Wang, S. Chen, "Esa-cyclegan: Edge feature and self-attention based cycle-consistent generative adversarial network for style transfer," *IET Image Processing*, vol. 16, no. 1, pp. 176–190, 2022, doi: https://doi.org/10.1049/ipr2.12342.

[12] Z. Yuan, M. Jiang, Y. Wang, B. Wei, Y. Li, P. Wang, W. Menpes-Smith, Z. Niu, G. Yang, "Sara-gan: Self-attention and relative average discriminator based generative adversarial networks for fast compressed sensing mri reconstruction," *Frontiers in Neuroinformatics*, vol. 14, p. 611666, 2020.

[13] L. Deng, "The mnist database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.

[14] M. Mirza, S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[15] J.-Y. Zhu, T. Park, P. Isola, A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[16] Y. Lu, Y.-W. Tai, C.-K. Tang, "Attribute-guided face generation using conditional cyclegan," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[17] G. Brunner, Y. Wang, R. Wattenhofer, S. Zhao, "Symbolic music genre transfer with cyclegan," in *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, 2018, pp. 786–793.

[18] G. Bargshady, X. Zhou, P. D. Barua, R. Gururajan, Y. Li, U. R. Acharya, "Application of cyclegan and transfer learning techniques for automated detection of covid-19 using x-ray images," *Pattern Recognition Letters*, vol. 153, pp. 67–74, 2022.

[19] M. Park, D. Q. Tran, D. Jung, S. Park, "Wildfire-detection method using densenet and cyclegan data augmentation-based remote camera imagery," *Remote Sensing*, vol. 12, no. 22, 2020, doi: 10.3390/rs12223715.

[20] H. Tang, H. Liu, D. Xu, P. H. S. Torr, N. Sebe, "Attentiongan: Unpaired image-to-image translation using attention-guided generative adversarial networks," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–16, 2021, doi: 10.1109/TNNLS.2021.3105725.

[21] Y. Lin, Y. Wang, Y. Li, Y. Gao, Z. Wang, L. Khan, "Attention-based spatial guidance for image-to-image translation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 816–825.

[22] D. Bahdanau, K. Cho, Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[23] T. Luong, H. Pham, C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, Sept. 2015, pp. 1412–1421, Association for Computational Linguistics.

[24] A. Kolesnikov, A. Dosovitskiy, D. Weissenborn, G. Heigold, J. Uszkoreit, L. Beyer, M. Minderer, M. Dehghani, N. Houlsby, S. Gelly, T. Unterthiner, X. Zhai, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[25] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 10012–10022.

[26] K. Yue, M. Sun, Y. Yuan, F. Zhou, E. Ding, F. Xu, "Compact generalized non-local network," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, Curran Associates, Inc.

[27] S. Zhang, X. He, S. Yan, "LatentGNN: Learning efficient non-local relations for visual recognition," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97 of *Proceedings of Machine Learning Research*, 09–15 Jun 2019, pp. 7374–7383, PMLR.

[28] Z. Shen, M. Zhang, H. Zhao, S. Yi, H. Li, "Efficient attention: Attention with linear complexities," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021, pp. 3531–3539.

[29] H. Zhang, I. Goodfellow, D. Metaxas, A. Odena, "Self-attention generative adversarial networks," in *International conference on machine learning*, 2019, pp. 7354–7363, PMLR.

[30] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[31] Y. Taigman, A. Polyak, L. Wolf, "Unsupervised cross-domain image generation," *arXiv preprint arXiv:1611.02200*, 2016.

[32] C. Sakaridis, D. Dai, L. Van Gool, "Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 10765–10775.

[33] D. P. Kingma, J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[34] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, vol. 29, 2016, Curran Associates, Inc.

[35] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, Curran Associates, Inc.

[36] M. Bińkowski, D. J. Sutherland, M. Arbel, A. Gretton, "Demystifying mmd gans," *arXiv preprint arXiv:1801.01401*, 2018.

[37] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[38] A. Obukhov, M. Seitzer, P.-W. Wu, S. Zhydenko, J. Kyl, E. Y.-J. Lin, "High-fidelity performance metrics for generative models in pytorch," 2020. [Online]. Available: https://github.com/toshas/torch-fidelity, doi: 10.5281/zenodo.4957738, Version: 0.3.0, DOI: 10.5281/zenodo.4957738.