THE HONG KONG POLYTECHNIC UNIVERSITY
香港理工大學

Department of Electrical and Electronic Engineering

Final Year Project Final Report (2023/24)

Generation of realistic 2D scenes by Text-to-2D Models

Student Name	**HAO Jiadong**
Student ID	**20084595D**
Programme	**46403**
Academic Supervisor	**Prof. LAM Kin Man**
Industrial Supervisor	**Mr. Wong Chun Sang**
Submission Date	**April 11, 2024**

The Hong Kong Polytechnic University

Department of Department of Electrical and Electronic Engineering

EIE4430 Honours Project

1. **Student Name:** ___HAO Jiadong____ (**Student No.:** 20084595d)

2. **Programme Code:** 46403

3. **Project Title:** Generation of realistic 2D scenes by Text-to-2D Models

4. **Supervisor Name:** Prof. LAM Kin Man

5. **Project summary**

Objective:

Improve AttnGAN in terms of textual understanding and efficiency

Results:

1. Developed a novel architecture called Trans_AttnGAN. Integrated a pre-trained BERT model for generating more contextually accurate sentences and word embeddings. Designed a novel Soft Alignment Loss, leveraging a pre-trained image captioning model (BLIP) followed by a BERT to generate fine-grained guidance in sentence and word level.

2. Conducted intensive experiments in fine-tuning and component analysis Trans_AttnGAN.

3. Verified that Trans_AttnGAN achieved comparable performance to AttnGAN with roughly half the total training time on the CUB-200 dataset.

DECLARATION OF ORIGINALITY

Except where reference is made in the text of this report, I declare that this report contains no material published elsewhere or extracted in whole or in part from any works or assignments presented by me or any other parties for another subject. In addition, it has not been submitted for the award of any other degree or diploma in any other tertiary institution.

No other person's work has been used without due acknowledgment in the main text of the Report.

I fully understand that any discrepancy from the above statements will constitute a case of plagiarism and be subject to the associated.

Signature

Abstract

This project introduces Trans-AttnGAN, a novel text-to-image generation model that significantly refines its predecessor, AttnGAN. With a transformer-based pre-trained model BERT, Trans-AttnGAN can achieve more nuanced and contextually accurate interpretation of textual descriptions, thereby enhancing the quality and semantic coherence of the generated images. In addition, a novel Soft Alignment Loss is employed to replace AttnGAN's computationally demanding Deep Attentional Multimodal Similarity Model (DAMSM), which greatly shrinks the training time of AttnGAN. In our experiment, we verify that Trans-AttnGAN achieves comparable performance to AttnGAN in roughly half the total training time on the CUB-200 dataset, marking a significant improvement in training efficiency.

Acknowledgment

I would like to express my heartfelt thanks to my supervisor, Dr. LAM Kin Man, who held regular meetings with me and offered valuable advice and support during my final year project. Besides, I would like to express my great appreciation to my industrial supervisor, Mr. Wong Chun Sang, who provided me with papers and codes for reference. I would also like to thank Mr. Hui Wai Lam for his time in arranging GPU and remote access to the server for me to train and test my model.

Table of Content

List of Tables

List of Figures

Chapter 1. Introduction

Text-to-image generation is a challenging task that encompasses topics like Natural Language Processing (NLP), Computer Vision (CV) and multimodality learning. This interdisciplinary field has many fantastic applications, such as digital art production and advertisement poster generation [1, 2, 3].

Many text-to-image models are based on the Generative Adversarial Networks (GANs) invented by I. Goodfellow et al. [4]. AttnGAN, proposed by T. Xu et al., is a milestone of such implementation. By designing an attentional generative network for multi-stage image generation and a Deep Attentional Multimodal Similarity Model (DAMSM) for fine-grained image-text matching loss, it has shown prominent capabilities to generate realistic images that can reflect detailed information in the text prompts [5].

While AttnGAN establishes a benchmark in text-to-image synthesis, it presents room for enhancement, particularly in the realms of textual understanding and training efficiency. In terms of textual understanding, AttnGAN employs Long Short-Term Memory (LSTM) [6] as its text encoder, generating sentence and word embeddings for the attention mechanism. However, LSTM networks have inherent limitations in capturing the more nuanced, context-dependent features of the language, especially compared to more advanced models like transformers [7], which may bottleneck the performance of AttnGAN. In terms of training efficiency, AttnGAN introduces a Deep Attentional Multimodal Similarity Model (DAMSM) to calculate a fine-grained loss to provide sentence and word-level guidance to the generator. However, DAMSM needs pre-training for at least 200 epochs, which is time-consuming and computationally demanding.

In this project, we propose trans_AttnGAN, effectively addressing the two challenges that AttnGAN is facing, further improving its overall performance. The overall architecture of the trans_AttnGAN is shown in Figure 1. Firstly, we replace the LSTM in AttnGAN with a pre-trained transformer model, BERT [8], to generate more contextually accurate sentence and word embeddings for the Attentional Generative Network. Furthermore, we replace the computationally intensive DAMSM Loss in AttnGAN with a lightweight Soft Alignment Loss. In this novel approach, a pre-trained image captioning model BLIP [9] is used to first transform the images back into descriptive captions. Subsequently, a pre-trained BERT transformer is employed to convert these generated captions into meaningful sentence and word features. Finally, cosine similarity scores between the real captions and the generated captions are calculated in both sentence and word levels to measure how closely the generated images align with the text prompts.

As we use pre-trained models, the proposed trans-AttnGAN requires less training time and computational resources while retaining the performance as compared to the AttnGAN. Our code is available at https://github.com/IUboyfriend/trans_AttnGAN.

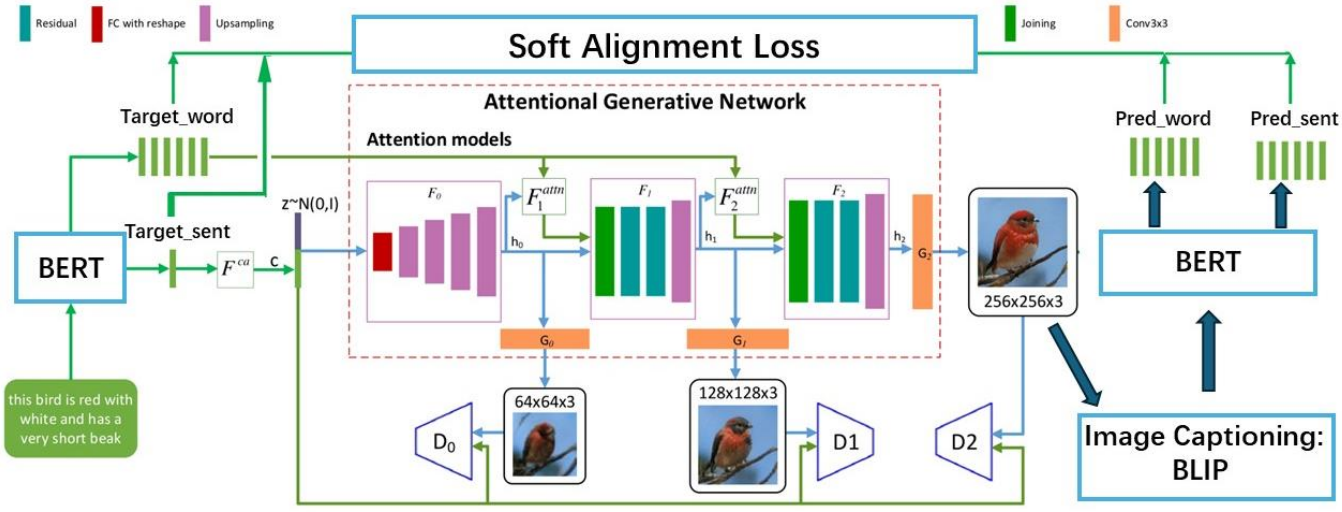


Figure 1. Architecture of the proposed trans_AttnGAN

Chapter 2. Related Work

2.1 Text-to-Image Generation

Generating high-quality images from text prompts is a challenging and complex task. It has achieved remarkable progress with the recent advancement of a variety of deep generative models. For example, Reed et al. used conditional PixelCNN to create images from text by a multi-scale model structure [10]. Mansimov et al. proposed alignDRAW, an extension of the Deep Recurrent Attention Writer (DRAW), which iteratively constructs image patches while focusing on relevant words in text descriptions [11]. Nguyen et al. introduced an approximate Langevin sampling method for text-conditioned image generation [12]. Among all the generative models, GANs are one of the major methods that have shown outstanding performance in text-to-image generation.

2.2 Generative Adversarial Networks (GANs)

In GANs, a generator (G) and a discriminator (D) are trained in an adversarial manner, improving themselves against each other progressively [4]. The generator (G) starts with random noise and tries to generate samples that resemble the real data distribution to fool the discriminator, while the discriminator (D) attempts to distinguish the real data and the fake data produced by the generator. The min-max objective function is modeled as $\min_G \max_D V(D,G) = E_{x \sim p_{data}(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))]$. $E_{x \sim pdata(x)}[\log D(x)]$ represents the expectation of the discriminator's estimates for real data x. The discriminator tries to maximize this term to show great confidence in its recognition and affirmation of real data. $E_{z \sim p_z(z)}[\log(1 - D(G(z)))$ represents the expectation of the discriminator's estimates for data produced by the generator. The generator tries to minimize this term, making the discriminator believe the generated samples are real, while the discriminator tries to maximize this term, having strong confidence in recognizing the fake data.

2.3 AttnGAN

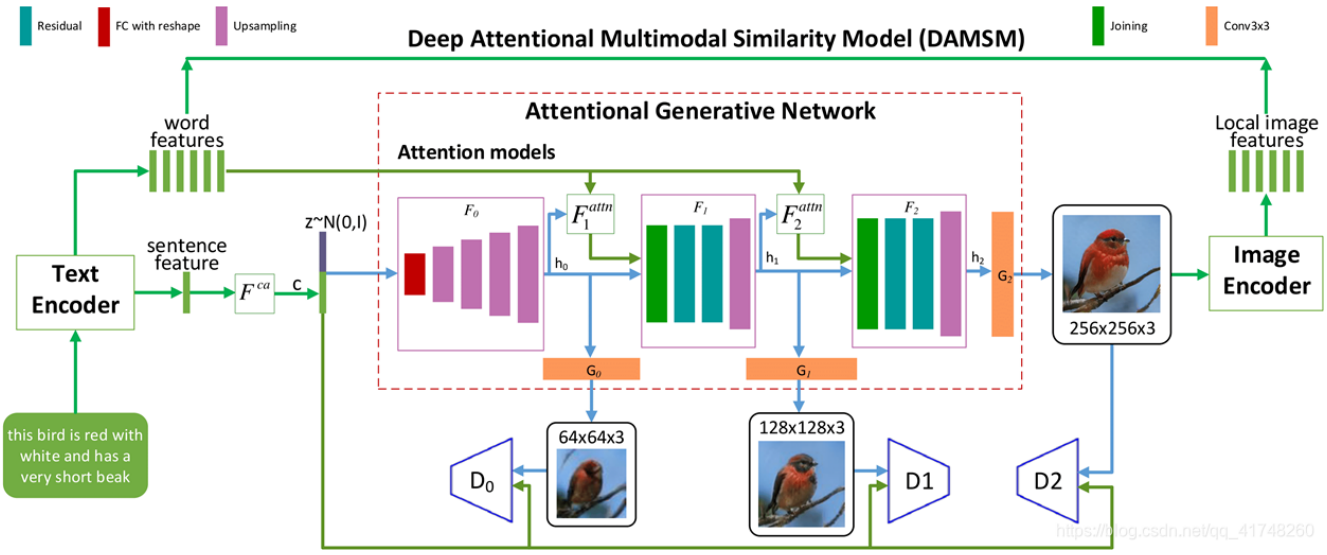


Figure 2. Architecture of AttnGAN

2.3.1 Attentional Generative Network

As shown in Figure 2, AttnGAN comprises multiple generators *(G0, G1, ..., Gm-1)*, generating images from a smaller scale to a larger scale. These generators are fed with the corresponding hidden states $(h_0, h_1, \ldots, h_{m-1})$.

To get the first hidden state h_0, a noise vector z is concatenated with the conditioning augmentation (F^{ca}) of a sentence vector $\bar{e}$, and then fed to the reshaping network F_0 (Eq.1).

$$h_0 = F_0(z, F^{ca}(\bar{e})); \quad (1)$$

Other hidden states h_i is computed from the last hidden states h_{i-1} and an attention-driven word-context matrix. The word context matrix is generated by the attention model F_i^{attn} fed by the word features e produced by the text encoder and the last hidden state h_{i-1} (Eq.2).

$$h_i = F_i(h_{i-1}, F_i^{attn}(e, h_{i-1})) \text{ for } i = 1, 2, ..., m-1; \quad (2)$$

For the j[th] sub-region (a column in the hidden state), its word-context vector c_j is calculated by (Eq.3), where e'_i is the reshaped feature vector of the i[th] word, $\beta_{j,i}$ is the attention weight indicating how much the model attends to the i[th] word when generating the j[th] sub-region.

$$c_j = \sum_{i=0}^{T-1} \beta_{j,i} e'_i, \text{ where } \beta_{j,i} = \frac{\exp(s'_{j,i})}{\sum_{k=0}^{T-1} \exp(s'_{j,k})}$$

$$s'_{j,i} = h_j^T e'_i \quad (3)$$

Finally, we combine the word-context vectors for N sub-regions to get the word-context matrix for the whole image (Eq.4):

$$F^{attn}(e,h) = (c_0, c_1, ..., c_{N-1}) \quad (4)$$

Once the hidden state is ready, it is fed to the generator G_i to generate the image of the corresponding scale. The generator loss and the discriminator loss are defined as (Eq.5,6), where x_i is from the true image distribution $p_{data\text{-}i}$ while $\hat{x}_i$ is from the model distribution p_{Gi}. The unconditional loss determines whether the image is real or fake, while the conditional loss determines whether the image and the sentence match or not.

$$\mathcal{L}_{G_i} = \underbrace{-\frac{1}{2}\mathbb{E}_{\hat{x}_i \sim p_{G_i}}[\log(D_i(\hat{x}_i)]}_{\text{unconditional loss}} \underbrace{-\frac{1}{2}\mathbb{E}_{\hat{x}_i \sim p_{G_i}}[\log(D_i(\hat{x}_i, \overline{e})]}_{\text{conditional loss}} \quad (5)$$

$$\mathcal{L}_{D_i} = \underbrace{-\frac{1}{2}\mathbb{E}_{x_i \sim p_{data_i}}[\log D_i(x_i)] - \frac{1}{2}\mathbb{E}_{\hat{x}_i \sim p_{G_i}}[\log(1 - D_i(\hat{x}_i)] +}_{\text{unconditional loss}}$$

$$\underbrace{-\frac{1}{2}\mathbb{E}_{x_i \sim p_{data_i}}[\log D_i(x_i, \overline{e})] - \frac{1}{2}\mathbb{E}_{\hat{x}_i \sim p_{G_i}}[\log(1 - D_i(\hat{x}_i, \overline{e})],}_{\text{conditional loss}} \quad (6)$$

2.3.2 Deep Attentional Multimodal Similarity Model (DAMSM)

The DAMSM consists of a text encoder (LSTM) and an image encoder (CNN) that maps the words and the images into a common space and then calculates the image-text similarity at the word and sentence level to give fine-grained guidance for image generation.

Firstly, the LSTM turns the caption into the sentence feature vector $\bar{e}$ and word feature matrix e, and the CNN turns the image into the global image feature vector $\bar{v}$ and the local image feature matrix v.

Subsequently, a similarity matrix s for all possible pairs of words and the image sub-regions is calculated and normalized $\bar{s}$, where i represents the ith word and j represents the jth sub-region of the image (Eq.7).

$$s = e^T v$$
$$\bar{s}_{i,j} = \frac{\exp(s_{i,j})}{\sum_{k=0}^{T-1} \exp(s_{k,j})} \quad (7)$$

Then, a region-context-vector for each word is computed as the weighted sum overall regional visual vector v_j (Eq.8).

$$c_i = \sum_{j=0}^{288} \alpha_j v_j, \quad \text{where } \alpha_j = \frac{\exp(\gamma_1 \bar{s}_{i,j})}{\sum_{k=0}^{288} \exp(\gamma_1 \bar{s}_{i,k})} \quad (8)$$

Finally, the relevance between the ith word and the whole image is represented by the cosine similarity between c_i and e_i (Eq.9). The attention-driven image-text matching score between the entire image Q and the whole text descript D is defined in (Eq.10).

$$R(\mathbf{c}_i, \mathbf{e}_i) = \frac{\mathbf{c}_i^T \mathbf{e}_i}{\|\mathbf{c}_i\| \|\mathbf{e}_i\|} \quad (9)$$

$$R(Q, D) = \log \Big(\sum_{i=1}^{T-1} \exp(\gamma_2 R(c_i, e_i)) \Big)^{\frac{1}{\gamma_2}} \quad (10)$$

In a batch of image-sentence pairs $\{(Qi, Di)\}_{i=1}^{M}$, only Di matches with Qi, and all other *M-1* sentences are treated as mismatching descriptions. Hence, we have two posterior probabilities of the sentence matching with the image (Eq.11):

$$P(D_i|Q_i) = \frac{\exp(\gamma_3 R(Q_i, D_i))}{\sum_{j=1}^{M} \exp(\gamma_3 R(Q_i, D_j))}$$

$$P(Q_i|D_i) = \frac{\exp(\gamma_3 R(Q_i, D_i))}{\sum_{j=1}^{M} \exp(\gamma_3 R(Q_j, D_i))} \quad (11)$$

Then the two word-level losses (L_1^w, L_2^w) are defined as (Eq.12):

$$\mathcal{L}_1^w = -\sum_{i=1}^{M} \log P(D_i|Q_i)$$

$$\mathcal{L}_2^w = -\sum_{i=1}^{M} \log P(Q_i|D_i) \quad (12)$$

If we redefine (Eq.10) by (Eq.13), where $\bar{v}$ is the global image feature vector and $\bar{e}$ is the global sentence feature vector, and substitute it into (Eq.11 and 12), we will have two sentence-level losses (L_1^s, L_2^s).

$$R(Q, D) = \frac{\bar{v}^T \bar{e}}{\|\bar{v}\| \|\bar{e}\|} \quad (13)$$

Finally, the DAMSM loss I defined as (Eq.14):

$$\mathcal{L}_{DAMSM} = \mathcal{L}_1^w + \mathcal{L}_2^w + \mathcal{L}_1^s + \mathcal{L}_2^s \quad (14)$$

The generator loss is defined as (Eq.15), where m is the number of the generators and λ is a hyperparameter weighting the importance of the DAMSM.

$$\mathcal{L} = \mathcal{L}_G + \lambda \mathcal{L}_{DAMSM}, \quad \text{where } \mathcal{L}_G = \sum_{i=0}^{m-1} \mathcal{L}_{G_i} \tag{15}$$

2.4 Inception Score for Evaluation

Inception Score is derived from a pre-trained Inception network and evaluates the generative model from two aspects [13]. The first aspect is the quality of individual images. The Inception network will predict the class label of each generated image where the classification confidence indicates their recognizability. The second aspect is the diversity of the entire set. It is measured by calculating the entropy of the class distributions. If most of the generated images are similar, the class distribution would have low entropy. Conversely, if the generated images are diverse, the entropy will be high.

Chapter 3 Trans-AttnGAN

3.1 Rationale and Development of Trans-AttnGAN Design

Initially, motivated by the advantages of the transformers over LSTMs in textual understanding and embedding generation, we aim to replace the text encoder of AttnGAN, which is a bi-directional LSTM, with a transformer BERT. However, in the experiment, we encountered significant challenges that led us to reconsider this scheme for the following reasons.

Firstly, changing to BERT leads to much more time and computational power consumption. In the original AttnGAN, pre-training of DAMSM merely requires 130.34ms (time for a batch) * 553 (number of batches in an epoch) = 72.08s for one epoch in average shown in Figure 3. However, if changing LSTM to BERT, we need 1181.39ms * 553 = 653.31s for one epoch shown in Figure 4, approximately 9 times more than the LSTM. The DAMSM needs at least 200 epochs of pre-training, as raised by the author of AttnGAN [5], which means pre-training a usable DAMSM will cost over 36 hours, making it very difficult to fine-tune the DAMSM with BERT.

Secondly, the validation loss of the DAMSM with BERT is significantly higher than that of the original architecture, as shown in Figures 3 and 4, in both sentence and word levels. We also tried to freeze previous layers or use a distilled version of the transformer; although we can halve the training time, the error is still higher than the original AttnGAN, which indicates that the transformer is not fit for the DAMSM architecture.

```
| epoch   0 |   200/  553 batches | ms/batch 130.34 | s_loss  2.55  2.59 | w_loss  2.69  2.60
```

Figure 3. Training time and validation loss of DAMSM with LSTM

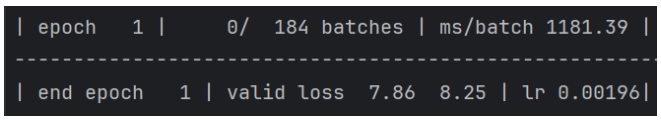


Figure 4. Training time and validation loss of DAMSM with BERT

However, changing the text encoder of AttnGAN to a transformer is still meaningful because the Attentional Generative Network leverages the sentence and word embeddings when the attention models produce the word-context vectors. BERT, known for its effectiveness in capturing contextual relationships within text, can produce more nuanced and contextually rich embeddings. Hence, we design the Trans_AttnGAN with a novel Soft Alignment Loss to replace the DAMSM to provide fine-grained sentence and word-level guidance. Trans_AttnGAN enables using BERT to generate more detailed and accurate sentence and word embeddings while discarding the pre-training stage that the original AttnGAN needs, greatly improving the performance and efficiency of AttnGAN.

3.2 Trans_AttnGAN Architecture

3.2.1 BERT text encoder

A pre-trained BERT named “bert-base-uncased” developed by Hugging Face [14] is used as the text encoder. It consists of 12 transformer blocks and 12 elf-attention heads, making it effective in finding deep semantic relationships and generating meaningful sentence and word embeddings.

The captions are first tokenized by a BERT tokenizer and padded to the longest sequence in the batch for efficient batch processing. Then, we freeze the BERT and feed the tokenized captions to it, which means the BERT is for inference only, not updating its pre-trained weights to save time

and computational resources. Next, we extract the last hidden state to obtain the sentence and word embeddings. We use the starting token [CLS] for sentence embeddings since it is trained through tasks like Next Sentence Prediction and encapsulates the entire sentence's context. The remaining embeddings, excluding the ending token [SEP] and the padding token, are considered word embeddings. Finally, we transform a batch of text captions into a tensor called *target_sent* (sentence embeddings) of shape [batch_size, hidden_size = 768] and a tensor called *target_word* (word embeddings) of shape [batch_size, hidden_size = 768, target_seq_length], where the *hidden_size* represents the dimensionality of embeddings in BERT and the *target_seq_length* represents the length of the longest captions in the batch.

3.2.2 Image Captioning Model BLIP and a Subsequent BERT

A pre-trained image captioning model BLIP named "Salesforce/blip-image-captioning-base" developed by Hugging Face [15] is employed to generate descriptive captions from the generated images before calculating the "Soft Alignment Loss". It effectively merges visual data with textual information using a combination of Vision Transformers and language models, tailored for generating accurate and context-aware captions.

During training, firstly, (Eq.16) is used to normalize the generated image from [-1,1] to [0,255], making them fit for the input picture format png. Then the normalized images are fed into a BLIP with all parameters frozen to generate descriptions of the fake images. Finally, the descriptions are input to a BERT to generate the sentence feature named *pred_sent* of size [batch_size, hidden_size = 768], and the word features named *pred_word* of size [batch_size, hidden_size = 768,

pred_seq_length], where the *pred_seq_length* represents the length of the longest generated captions in the batch.

$$\text{normalized\_img} = (img + 1.0) \times 127.5 \quad (16)$$

3.2.3 Soft Alignment Loss

Sentence-Level loss:

First, the two sentence embeddings, the *target_sent* and *pred_sent,* undergo L2 normalization to have norms equal to 1 so that the norms can be neglected in the denominator of the cosine similarity. Then, the cosine similarity and the sentence-level loss are calculated, as shown in (Eq.17).

$$\begin{aligned} \text{cos\_sim\_sent} &= \text{pred\_sent} \cdot \text{target\_sent} \\ \text{sent\_loss} &= 1 - \text{mean}(\text{cos\_sim\_sent}) \end{aligned} \quad (17)$$

Word-Level loss:

The word-level loss intends to check whether specific words in the target caption are reflected in the generated captions. This is crucial for ensuring that key features in the prompts are captured in the generated images.

Similar to the sentence-level loss, the two word-level embeddings *target_word* and *pred_word* first undergo the L2 normalization. The attention masks are then applied to exclude all irrelevant tokens like ending and padding tokens. Next, batch matrix multiplication is performed to compute a cosine similarity matrix *cos_sim_word* (Eq.18). The shape of this similarity matrix is [batch_size, pred_seq_len, target_seq_len].

$$\text{cos\_sim\_word} = \text{pred\_word} \cdot \text{target\_word} \quad (18)$$

Then we apply a Softmax function along the *pred_seq_len* dimension of the similarity matrix (Eq.19). The max value in each row is selected as the confidence score (Eq.20), aiming to find the most similar word in the *pred_seq_len* for each word in the *target_seq_len*.

$$\text{softmax\_scores} = \text{softmax}(\text{cos\_sim\_word}, \text{dim} = 1) \quad (19)$$

$$\text{max\_confidence\_scores} = \max(\text{softmax\_scores}, \text{dim} = 1) \quad (20)$$

Finally, these confidence scores are summed (Eq.21) and averaged (Eq.22) for each target caption. The word-level loss is modeled as (Eq.23).

$$\text{sum\_scores} = \text{sum}(\text{max\_confidence\_scores}, \text{dim} = 1) \quad (21)$$

$$\text{mean\_scores} = \frac{\text{sum\_scores}}{\text{pred\_cap\_len}} \quad (22)$$

$$\text{word\_loss} = \frac{\text{batch\_size} - \text{sum}(\text{mean\_scores})}{\text{batch\_size}} \quad (23)$$

The objective function of the proposed Trans_AttnGAN's attentional generative network is defined as (Eq.24), where $\alpha \; and \; \beta$ are two hyperparameters determine the relevant importance of the two proposed losses, and λ is the hyperparameter determines the relevant importance of the proposed Soft Alignment Loss.

$$L = \sum_{i=0}^{m-1} L_{G_i} + \lambda \left(\alpha \cdot \text{sent\_loss} + \beta \cdot \text{word\_loss} \right) \quad (24)$$

Chapter 4 Experiment

4.1 Dataset

In our experiment, we use the Caltech-UCSD Birds-200-2011 (CUB) dataset, which contains 11788 images (8,855 for training and 2,933 for testing) of 200 bird species [16]. Each image is assigned ten captions as the descriptions. For example, figures 5 and 6 show the captions and the corresponding image for the first sample in the class “Black_footed_Albatross”.

```
the medium sized bird has a dark grey color, a black downward curved beak, and long wings.
the bird is dark grey brown with a thick curved bill and a flat shaped tail.
bird has brown body feathers, white breast feathers and black beak
this bird has a dark brown overall body color, with a small white patch around the base of the bill.
the bird has very long and large brown wings, as well as a black body and a long black beak.
it is a type of albatross with black wings, tail, back and beak, and has a white ring at the base of its beak.
this bird has brown plumage and a white ring at the base of its long, curved brown beak.
the entire body is dark brown, as is the bill, with a white band encircling where the bill meets the head.
this bird is gray in color, with a large curved beak.
a large gray bird with a long wingspan and a long black beak.
```

Figure 5. Captions for the first sample in the class “Black_footed_Albatross.”

Figure 6. The first image in class “Black_footed_Albatross.”

4.2 Fine-Tune the BLIP Image Captioning Model

Though pre-trained on a large dataset, the BLIP fails to generate descriptive captions for the bird images. To generate descriptions that can capture the key features of the generated images, fine-tuning is needed. Because the BLIP has comprehensive prior knowledge, only minor efforts are

needed in fine-tuning. In our experiment, we try different learning rates, batch sizes and number of training samples. We find that 400 batches (batch size = 8) with learning rate = 1e - 6 is enough to gain a satisfying BLIP, which saves much time compared to the 200-epoch pre-training of the DAMSM of the AttnGAN. The comparison of the original BLIP and the fine-tuned BLIP is shown in Table 1. The results of the fine-tuned BLIP are shown in Table 2.

	**Without fine-tuning**	**With fine-tuning**
	"A small bird perched on a branch of a tree."	"This bird has wings that are brown and a white belly."

Table 1. Comparison of the BLIPs with and without fine-tuning

Image	**Real Caption**	**Generated Caption**
	"The bird has a **grey** side and breast as well as a **brown crown**."	"A small bird with a **grey** belly and a **brown crown**."
	"The bird has a **brown** body and a **pointed break** which color brighter than the rest of the body."	"This bird has wings that are **brown** and has a long **pointy beak**."

	"This is a small dirty **yellow** and **brown** bird with a **red and brown crown**."	"This bird has wings that are **brown** and **yellow** and has a **red crown**."

Table 2. Comparison between the real captions and captions generated by the fine-tuned BLIP

4.3 Component Analysis

In this section, we quantitatively evaluate Trans_GAN variants with the Inception Score to define the best sets of hyperparameters. The results are shown in Table 3. To save time, we only trained all the variants for 100 epochs.

Hyperparameters		**Inception Score**
$\lambda = 0$	-	3.78 ± 0.03
$\lambda = 10$	**$\alpha = 0.7, \beta = 0.3$**	**3.98 ± 0.07**
$\lambda = 10$	$\alpha = 0.5, \beta = 0.5$	3.86 ± 0.06
$\lambda = 10$	$\alpha = 0.3, \beta = 0.7$	3.83 ± 0.04
$\lambda = 50$	$\alpha = 0.7, \beta = 0.3$	3.74 ± 0.05
$\lambda = 50$	$\alpha = 0.5, \beta = 0.5$	3.69 ± 0.02
$\lambda = 50$	$\alpha = 0.3, \beta = 0.7$	3.67 ± 0.02

Table 3. Hyperparameter fine-tuning of Trans_AttnGAN (100 epochs)

From the above table, we can verify that our proposed Soft Alignment Loss has a positive impact on the quality of the generated image. This is because the parameter λ controls the relevant

importance of the Soft Alignment Loss to the total loss. When it equals zero, the inception score is lower than that of the experiment groups with $\lambda = 10$. Moreover, we can identify the best sets of hyperparameters, which are $\lambda = 10$, $\alpha = 0.7$, $\beta = 0.3$ for Trans_AttnGAN. We trained the TransGAN with the above set of hyperparameters for 200 epochs and got an inception score of **4.28 ± 0.03**.

4.4 Qualitative Evaluation of Trans_AttnGAN

In this section, we visualize the results of the best Trans_AttnGAN ($\lambda = 10$, $\alpha = 0.7$, $\beta = 0.3$, epoch = 200) to demonstrate its working flow and capacity.

Table 4. shows the outputs of the three generators of Trans_AttnGAN, visualizing how Trans_AttnGAN generates images by gradually refining the lower-resolution version to a higher resolution.

Caption: This bird has wings that are blue and black and has a red belly.		
		

Table 4. Three-level generation results of Trans_AttnGAN

Table 5. shows how the performance of Trans_AttnGAN improves throughout training epochs.

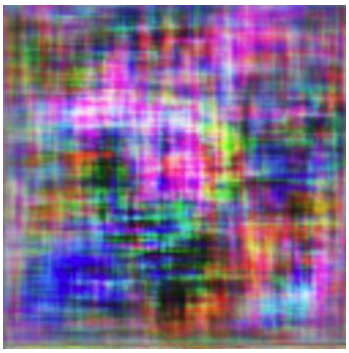
Epoch 1	Epoch 25	Epoch 50
		
Epoch 100	**Epoch 150**	**Epoch 200**
		

Table 5. Results of Trans_AttnGAN throughout training epochs

Table 6. shows the Trans_AttnGAN is able to generate diverse images adapting to the most attended words in the captions.

The bird has **wings** that are blue and has a red **belly**.				
				
The bird has **wings** that are black and has a white **belly**.				

				
The bird has **wings** that are grey and a yellow **belly**.				
				

Table 6. Results of Trans_AttnGAN when changing the most attended words

4.5 Comparison with AttnGAN

We compare Trans_AttnGAN with the best AttnGAN (λ =5, epoch = 600) mentioned in the paper [5]. Qualitatively, we show the first four outputs of the Trans_AttnGAN and the AttnGAN for the same input in Table 7.

Trans_AttnGAN		**AttnGAN**	
This bird is **black and white** and has a **very short beak**.			
			

Table 7. Comparison of Trans_GAN and AttnGAN

Table 7 reveals that, despite the presence of some unsuccessful images for both models, the

majority of images are realistic and rich in detail. Notably, both models effectively interpret and incorporate important textual prompts, like “long tail” or “red crown”, into their generated images, even in the failed generations. The comparative analysis from this sample set indicates that Trans_GAN’s performance is on par with that of AttnGAN.

Quantitatively, the Inception score of the best AttnGAN mentioned in the paper [5] is 4.36 ± 0.03, which is slightly better than our proposed Trans_AttnGAN’s 4.28 ± 0.03. However, as Table 8. shows, the AttnGAN is trained for 600 epochs with 200 epochs of pre-training of DAMSM, which requires 200 * 72.08 = 14,416s for pre-training DAMSM and 600 * 138 * 5.97 = 494,316s for training, in total 14,416 + 494,316 = 508,732s (approximately 6 days). In comparison, the proposed Trans_AttnGAN requires 400 * 18.12 = 7,248s for fine-tuning the BLIP and 200 * 138 * 9.50 = 262,200s for training, in total 7,248 + 262,200 = 269,448 s (approximately 3 days), which sees a significant increase in the training efficiency. Due to the time limitation, we do not train Trans_AttnGAN for 600 epochs. However, as shown in Figure 7, there is an increasing trend of the inception score of the Trans_AttnGAN, indicating that Trans_AttnGAN is very likely to outperform AttnGAN if training for more epochs.

Model	Pre-training (batch size = 16) (553 batch/epoch) (RTX 4060)	Training (batch size = 64) (138 batch/epoch) (RTX 3090)	Final Inception Score
Trans_AttnGAN	400 batches (18.12s/batch)	200 epochs (9.50s/batch)	4.28 ± 0.03

AttnGAN	200 epochs (72.08s/ epoch)	600 epochs (5.97s/batch)	4.36 ± 0.03

Table 8. Training time comparison between the AttnGAN and Trans_AttnGAN

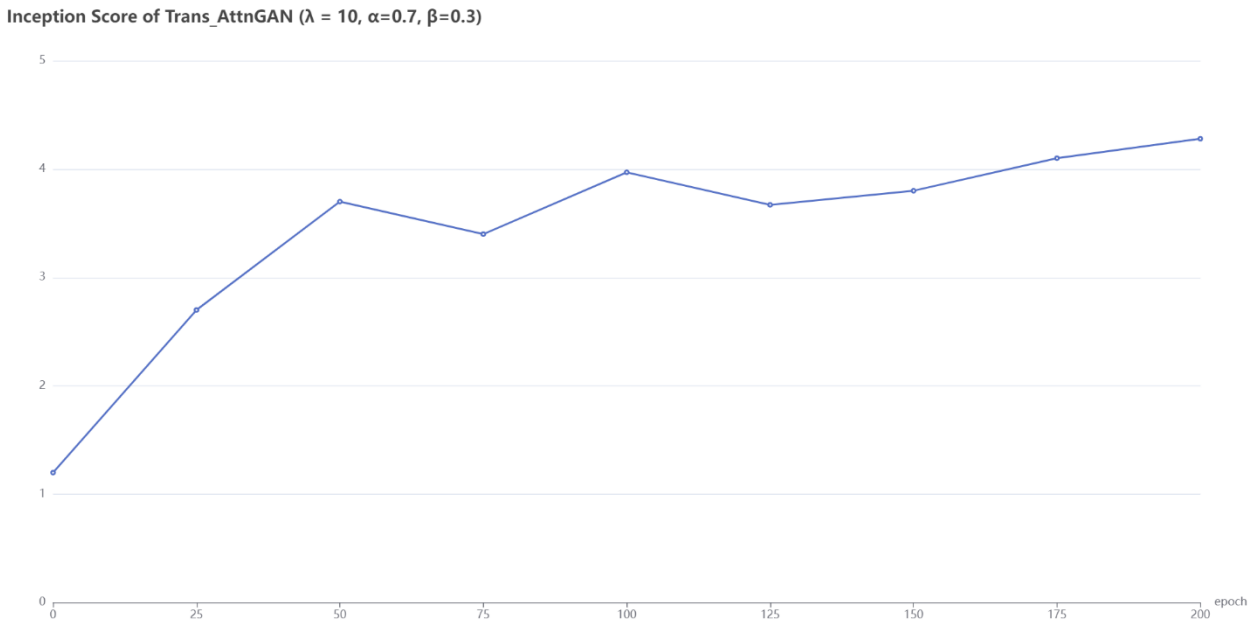


Figure 7. Inception Score of Trans_AttnGAN throughout epochs (sampled every 25 epochs)

Chapter 5 Conclusion

In this final year project, we introduced Trans-AttnGAN, a revised version of AttnGAN for text-to-image generation. The core innovation of Trans_AttnGAN employs BERT for enhanced textual understanding and a novel Soft Alignment Loss to replace the complex Deep Attentional Multimodal Similarity Model (DAMSM), thus greatly improving the training efficiency while maintaining high-quality image generation. Impressively, Trans_AttnGAN requires only about half the total training time compared to AttnGAN to achieve similar results. This reduction in training duration is a substantial improvement, especially in resource-constrained environments. If given enough time, Trans_AttnGAN is very likely to outperform AttnGAN since BERT can provide more fine-grained textual guidance to the generators than LSTM.

While we celebrate the successes of Trans_AttnGAN, we also recognize potential areas for enhancement. First, the Soft Alignment Loss focuses on the “single word” level, which may ignore important “phrase-level” information. For example, in a caption like “the bird has a red head”, the Soft Alignment Loss only checks whether the word “red” is reflected in the generated image but doesn’t explicitly ensure that the red color is attributed to the bird's head. Although our experiments demonstrate that the Trans_AttnGAN often successfully generates images with such details thanks to the descriptive sentence embedding generated by the BERT, the Soft Alignment Loss can be refined to provide more accurate “phrase-level” guidance. Secondly, currently we can only generate images of resolution 256 * 256, which is relatively low for practical use. If having time and enough computational power, we can stack more levels of generators to achieve a higher resolution.

Chapter 6 Reference

[1] H. Zhang et al., “Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks,” 2017 IEEE International Conference on Computer Vision (ICCV), Oct. 2017. doi:10.1109/iccv.2017.629

[2] S. Reed, Zeynep Akata, X. Yan, Lajanugen Logeswaran, B. Schiele, and H. Lee, “Generative Adversarial Text to Image Synthesis,” arXiv (Cornell University), May 2016.

[3] S. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," in Proceedings of the Advances in Neural Information Processing Systems (NIPS), 2016.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," in Proceedings of the IEEE Conference on Neural Information Processing Systems (NIPS), 2014, pp. 2672-2680.

[5] T. Xu et al., “AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks,” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Jun. 2018.

[6] Sepp Hochreiter and Jürgen Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[7] Ashish Vaswani et al., "Attention Is All You Need," in Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS), 2017.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, 2019.

[9] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Jiasen Lu, and Jianfeng Gao, "BLIP: Bootstrapped Language Image Pretraining for Vision-Language Tasks," arXiv preprint arXiv:2108.04473, 2021.

[10] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel Recurrent Neural Networks," in Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 2016, pp. 1747–1756.

[11] E. Mansimov, E. Parisotto, L. J. Ba, and R. Salakhutdinov, "Generating Images from Captions with Attention," in International Conference on Learning Representations (ICLR), San Juan, Puerto Rico, May 2016.

[12] A. Nguyen, J. Clune, Yoshua Bengio, Alexey Dosovitskiy, and J. Yosinski, “Plug & Play Generative Networks: Conditional Iterative Generation of Images in Latent Space,” arXiv (Cornell University), Nov. 2016.

[13] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved Techniques for Training GANs," in Proceedings of the Advances in Neural Information Processing Systems (NIPS), Barcelona, Spain, December 2016, pp. 2234–2242.

[14] Hugging Face. BERT-Base, Uncased [Online]. Available: https://huggingface.co/bert-base-uncased

[15] L. Li, Z. Lu, J. Lu, and H. Li, "BLIP: Bootstrapped Language Image Pretraining for Vision-Language Foundation Models," arXiv preprint arXiv:2201.12086, Jan. 2022.

[16] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," California Institute of Technology, CNS-TR-2011-001, 2011.