

Received 20 November 2023, accepted 11 December 2023, date of publication 14 December 2023,  
date of current version 22 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3342866

## RESEARCH ARTICLE

# GACnet-Text-to-Image Synthesis With Generative Models Using Attention Mechanisms With Contrastive Learning

MD. AHSAN HABIB<sup>1</sup>, MD. ANWAR HUSSEN WADUD<sup>1,2</sup>, LUBNA YEASMIN PINKY<sup>1</sup>,  
MEHEDI HASAN TALUKDER<sup>1</sup>, MOHAMMAD MOTIUR RAHMAN<sup>1</sup>,  
M. F. MRIDHA<sup>3</sup>, (Senior Member, IEEE), YUICHI OKUYAMA<sup>4</sup>, (Member, IEEE),  
AND JUNGPIL SHIN<sup>4</sup>, (Senior Member, IEEE)

<sup>1</sup>Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Tangail 1902, Bangladesh

<sup>2</sup>Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka 1216, Bangladesh

<sup>3</sup>Department of Computer Science, American International University-Bangladesh, Dhaka 1229, Bangladesh

<sup>4</sup>School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu 965-0006, Japan

Corresponding author: Jungpil Shin (jpshein@u-aizu.ac.jp)

**ABSTRACT** The generation of high-quality images from textual descriptions is a challenging task in computer vision and natural language processing. The goal of text-to-image synthesis, a current topic of research, is to produce excellent images from written descriptions. This study proposes a hybrid approach to evaluating a dataset consisting of various text-image pairs by efficiently combining conditional generative adversarial networks (C-GAN), attention mechanisms, and contrastive learning (C-GAN+ATT+CL). We suggest a two-step method to improve image quality that starts by utilizing generative adversarial networks (GANs) with attention mechanisms to create low-resolution images and then contrastive learning to improve. Contrastive learning modules train on a separate dataset of high-resolution pictures; GANs learn on datasets of low-resolution text and image pairs. The Conditional GAN with Attention Mechanism and Contrastive Learning Method provides state-of-the-art performance in terms of image quality, diversity, and visual realism, among the several methods. The results of this study demonstrate that the proposed approach works better than all other methods, achieving an Inception Score (IS) of 35.23, a Fréchet Inception Distance (FID) of 18.2, and an R-Precision of 89.14. Our findings demonstrate that our “C-GAN+ATT+CL” approach significantly improves image quality and diversity and offers exciting paths for further study.

**INDEX TERMS** Text-to-image synthesis, generative adversarial networks, C-GAN, attention mechanism, contrastive learning technique, consistency.

## I. INTRODUCTION

Text-to-image is a challenging task because it requires the model to understand the intricate relationships between different objects and their spatial arrangement in the image. Generative adversarial networks (GANs) have produced promising results in producing genuine pictures from noisy vectors. They are composed of a discriminator and a generator, two artificial neural networks that were trained

The associate editor coordinating the review of this manuscript and approving it for publication was Maria Chiara Caschera<sup>1</sup>.

against an adversary. The generator network attempts to produce visuals that can fool the discriminator, while the network of discriminators learns to distinguish real images from manufactured ones. Applying GANs to text-to-image creation is still challenging since the data being processed is now composed of text compared to a noise vector. Conditional GANs [1] (cGANs) and attention techniques are two remedies put forth by researchers to address this issue. A modification of the original GAN architecture is the conditional GAN (cGAN), which employs extra information to regulate the generator, like class labeling or

linguistic descriptions. In text-to-image generation, cGANs use a written explanation as a prerequisite for creating the corresponding image. It has been shown that this approach produces more convincing and coherent pictures than traditional GANs. C-GAN has been applied to a number of tasks, such as creating accurate pictures of faces, flowers, and birds from textual descriptions.

The attention mechanism [2], which allows the generator to concentrate on particular portions of the picture while creating it, is another crucial method for enhancing text-to-image generation. Attention processes enhance the standard of generated pictures and guarantee that they more closely match the written description. Each word or feature in the input text is given a weight by the attention mechanism, indicating its relative significance in creating the matching image. The most pertinent textual passages for producing the related image are taught to the attention mechanism along with the generator network during training. This is accomplished by tuning the generator network to minimize both the adversarial loss, which motivates the generated images to be indistinguishable from genuine images, and the attention loss, which motivates the attention mechanism to concentrate on the most important portions of the text.

Researchers have recently looked into the application of contrastive learning [3] for text-to-image creation. Contrastive learning is a method for learning concepts by comparing samples that are similar and those that are different. Contrastive learning can be employed in the context of text-to-image reconstruction to discover the connection between the written description and the created image. Contrastive learning, on the other hand, uses positive and negative pairs to contrast representations as a way of learning them. Object identification, picture retrieval, and image synthesis are just a few of the computer vision applications that have been demonstrated to be effective in learning representations. The effectiveness of C-GAN, attention mechanisms, and contrastive learning in text-image synthesis is examined in this article.

In this research, we propose a unique method for text-to-image fusion that combines cGANs, attention mechanisms, and contrastive learning. To encourage the image that is generated to appear more closely matched with the text outline, our method extends the cGAN model to include mechanisms for attention and contrastive loss. We assess our method using a number of benchmark datasets and contrast it with cutting-edge text-to-image generation strategies. The objectives of this paper are:

- To investigate the effectiveness of GANs in generating high-quality images from text.
- To generate text-image synthesis methods combining C-GAN, attention mechanisms, and contrastive learning techniques.
- To assess the effectiveness of the proposed method with text embedding techniques pre-trained, BERT in text-image synthesis.

- To assess the effectiveness of contrastive learning with loss functions like cosine similarity, contrastive loss is useful for text-image synthesis.

Following is the breakdown of the remaining sections of this paper: In Section II, a literature review and related materials are offered in detail. In Section III, the research methodology and key discussion points are presented. In Section IV, the results and key discussion points are displayed. Section V concludes with a statement.

## II. RELATED WORKS

The field of text-to-picture synthesis has experienced significant contributions from many outstanding people. Several synthetic image applications, including super-resolution, picture generation, and images in paintings, have made use of GANs. In the context of text-image manufacturing, GANs have demonstrated their efficacy in generating conceptually consistent pictures based on a given text's contents. On the Caltech-UCSD Birds-200 (CUB) dataset, researchers provide TextControlGAN, a controllable GAN-based model that achieves a 17.6% improvement in Inception Score (IS) and a 36.6% reduction in Fréchet Inception Distance (FID) [4]. The contributors to the paper [5] proposed the use of an Attentional Generative Adversarial Network (AttnGAN), which worked for multi-stage, attention-driven refining for precise text-to-image generation. The best-reported inception score increased by 14.14% on the CUB dataset and 17.25% on the more difficult COCO dataset. Researchers in the paper [6] developed the MirrorGAN text-to-image-to-text architecture as an innovative, global-locally attentive, and semantic-preserving solution to this issue. Extensive tests on two publicly available benchmark datasets showed that MirrorGAN was superior to other representative state-of-the-art approaches. A brand-new framework called the Generative Adversarial What-Where Network (GAWWN), which created graphics based on instructions indicating what should be drawn where [7] With the Caltech-UCSD Birds dataset, they demonstrated high-quality  $128 \times 128$  image synthesis that was dependent on both informal text descriptions and object position. Their technique had made the management of the bounding box surrounding the bird and its individual components visible. To enhance the standard and semantic reliability of generated pictures, conditional GAN (cGAN) was developed. As an illustration [8], the authors of the paper recommended using contrastive learning to improve the appearance and uniformity of synthetic images. The researchers in [9] suggested an approach based on the recently announced Implicit Maximum Likelihood Estimation (IMLE) framework, in contrast to the majority of older methods, which utilized the GAN architecture. The generating network created visuals while concentrating on different parts of the textual description and utilizing the attention method. For instance, [10] suggested an attention-based cGAN model for text-image synthesis that excelled on the CUB dataset [11]. In this research, researchers developed the Self-Attention

Generative Adversarial Network (SAMGAN), which allowed attention-driven, long-range dependency modeling for image creation issues. They suggested SAMGAN raise the best published inception score from 36.8 to 52.52 while lowering the Fréchet inception distance on the challenging ImageNet dataset from 27.62 to 18.65. Several recent works applied contrastive learning to text-image synthesis using GAN and obtained results at their peak. The paper [12] proposed a novel GAN text-image synthesis method called CLIP-guided contrastive learning that made use of contrastive learning. The basic idea behind their method was to use a contrastive learning objective to align image and text integrations created by a pre-trained visual language model called CLIP. To compare the positive and negative pairs of embeddings, they explicitly used the CLIP model to independently encode generated images and text descriptions. They showed that their solution beat cutting-edge methods for several benchmark data sets. Additional recent research has looked into employing GANs and adversarial learning for text-image synthesis. For instance, [13] proposed GAN + LSTM + Attention for text-image synthesis on benchmark data sets such as MNIST, CIFAR-10, and SVHN. Another research article [14] found that DF-GAN was simpler, more efficient, and yielded better results. Extensive experimental and ablation investigations showed that the proposed model was superior to existing models for the Caltech-UCSD Birds 200 and COCO datasets. The contributors of [15] introduced a unique method for text-image synthesis using GAN that aligns text and image attributes using adversarial learning. The main strategy was to combine GAN with semantic attention. The people involved [16] developed a GAN-based method that created images from text descriptions by combining semantic segmentation and object detection. Both quantitative and qualitative metrics showed that the suggested strategy performed better than current GAN-based approaches. The main principle of their method was to enhance the quality and diversity of the generated images by using a semantic consistency loss in addition to a contrastive learning loss. They specifically encoded the verbal description and the generated image independently using a pre-trained vision-language model called CLIP and then compared the positive and negative pairs of image-text embeddings. One of the main problems with GANs was how few different kinds of images they could produce. Contrastive learning has often been used in work to encourage variation in the generated images in an effort to address this problem. For instance, [17] authors evaluated their approach to two renowned text-to-image generation methods, AttnGAN and DM-GAN, using data sets such as CUB and COCO. Experimental results showed that their approach could successfully improve the standard of synthetic images through the use of the three metrics of IS, FID, and R-precision. In the following stage of GAN training, they also employed the contrastive learning strategy to increase the reliability of the generated images based on the captions for the same picture. Among this literature, we have focused

on recent attempts that give a conception of combining GANs with contrastive learning for text-to-picture synthesis. We've examined several articles that used contrastive learning to get over some of the limitations of GANs' text-to-image synthesis, like the mode collapsing issue and the lack of variety among the output pictures. Additionally, we have discussed the many techniques used to improve the diversity and alignment of the created images using contrast learning.

### III. METHODOLOGY

In this part, we outline our approach for the text-to-image synthesis technique and suggest utilizing C-GAN with contrastive learning and an attention mechanism in Figure 1. Preprocessing, text encoder, image encoder, image generation, contrastive learning, and training procedures make up the methodology section. Significant information on methodology:

- A generator, a discriminator, an attention mechanism, and a contrastive loss component make up our suggested model.
- The text encoder has to transform the input text into its latent vector representation. The image generator uses this latent vector as its input to produce the corresponding image.
- A conditional generative adversarial network (C-GAN) architecture is used for training the generator.
- The generator model creates a related image using a textual description and a random noise vector as input.
- An image and textual description are inputs into the discriminator model, which then generates a probability score indicating whether the image-text pair is authentic or fraudulent.
- When creating the image, a soft attention mechanism based on the model is employed to direct attention to the pertinent sections of the textual description.
- One of the factors in the attention mechanism is the hidden state's dimensions, along with the number of attention heads and the dimensions of the attention output.
- The InfoNCE loss is used to implement the contrastive learning loss, which encourages the generator to create a variety of visually unique images given various textual descriptions.

#### A. PREPROCESSING AND DATA ANALYSIS

For every machine learning activity, including text-image synthesis employing GANs and contrastive learning approaches, data pretreatment and analysis are crucial phases. We convert raw data into a format suited for analysis and training. Here, along with the corresponding mathematical formulations, we present some of the key data pretreatment and analysis procedures employed in this effort.

##### 1) TEXT PREPROCESSING

Text data must be preprocessed to extract useful information before it can be used to create images. To put text into a

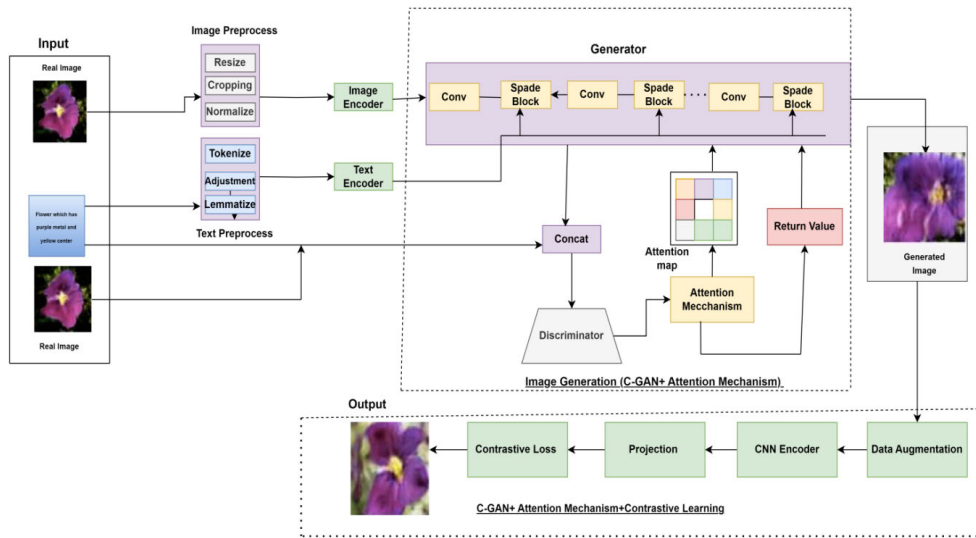


FIGURE 1. Combination of C-GAN+CL+ATT (Proposed Model).

more uniform format, it goes through processes including tokenization, adjustment, and lemmatization.

### 2) TOKENIZATION

Tokenization is the process of separating text into tokens, such as words, which are then utilized as input for additional analysis. Text is typically broken up into spaces, punctuation, and other separators as a standard tokenization technique. Heres an example of a string of data: “Yellow stamen and a white flower”. With tokenization, wed get something like this: “yellow,” “stamen,” “and,” “a,” “white,” “flower”.

### 3) LEMMATIZATION

This helps standardize the supplied data by transforming terms to their root or dictionary form. The WordNet lemmatize is a popular method for lemmatization that makes use of a database of word forms and their related base forms. The word “walk,” for instance, may be spelled “walking,” “walks,” or “walked.” The letters “s,” “ed,” and “ing” that indicate inflection are eliminated. These words are grouped together by their common lemma, “walk”.

### 4) PRE-PROCESSING IMAGES

Images must be pre-processed to standardize their size and color distribution before we can utilize them as training input. To transform an image into a more uniform format, it entails actions like scaling, cropping, and normalizing.

### 5) RESIZING

This helps to minimize the dimensionality of the supplied data by scaling the image to a standard size. Scaling an image to a specific width or height is a typical method of resizing. By selecting a specific area of interest inside an image, we can get rid of extraneous background details. Selecting a rectangular zone of interest inside an image is a typical cropping technique.

### 6) NORMALIZATION

A process that normalizes the color spectrum of an image to lessen the impact of lighting and other environmental factors? Subtracting the average color value of each pixel and dividing by the standard deviation is a typical normalizing technique.

For text-image synthesis utilizing GAN and contrastive learning approaches, data pretreatment and analysis are typically essential steps. We can make sure that our models can discover significant patterns and provide high-quality photos by transforming the raw data into a consistent format.

### B. TEXT ENCODER

Initially, textual representations are encoded into feature representations that the generator networks can employ. The text is encoded using a bidirectional LSTM network of neurons and then put into a mechanism for attention. The attention mechanism gives the input sequence weights so that the model can concentrate on the most valuable data. In text-image synthesis utilizing GANs and contrastive learning, the input text is encoded into a fixed-length vector representation using a text encoder. The text encoder, which transforms the input text into a place where it can be easily compared to embedded images for contrast loss, is critical to the overall performance of the model. Using a recurrent neural network (RNN), such as an LSTM or GRU, is a typical way to encode text. RNNs create a hidden state for each character in the text input as a series of characters. After that, text embedding can be done using the final masked state. The following equation provides the calculation of the hidden state at each time step t:

$$z_t = W_z * x_t + U_z * h_{t-1} + b_z \tag{1}$$

$$r_t = W_r * x_t + U_r * h_{t-1} + b_r \tag{2}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tanh(W_h * x_t + U_h * (r_t * h_{t-1}) + b_h) \tag{3}$$

where  $W_z, U_z, b_z, W_r, U_r, b_r, W_h, U_h, b_h$  are learnable parameters.



The final hidden state  $h_n$  can be used as text embedding, i.e.,  $e = h_n$ .

For text encoding, we can also utilize a transformer-based design like BERT [18]. In the context of text-to-image synthesis, BERT embedding techniques involve using BERT to encode textual descriptions or captions associated with images. The goal is to obtain meaningful and contextual representations of the text that can be used as input for the image synthesis model. BERT embeddings play a crucial role in bridging the gap between text and image modalities. These embeddings capture the contextual information and semantic representations of the input text, enabling the model to understand and generate corresponding visual content.

Convolutional neural networks (CNNs) [19] are frequently employed as the image encoder in GANs. These networks take an input image and output a feature vector, which is fed into the generator or discriminator. A CNN is composed of a number of convolutional layers with filters, an activation function such as ReLU or LeakyReLU, and a layer for pooling or downsampling data such as MaxPooling. Before being employed in the generator or discriminator, the output feature vector is typically flattened and routed through one or more completely linked layers. The image encoder's mathematical formulation can be stated as follows: Let  $I$  be the input image, and  $W$  and  $b$  be the weight matrix and bias vector of the convolutional layer, respectively. Then, the output feature vector  $F$  can be obtained as follows:

$$F = f(W * I + b) \tag{4}$$

where  $*$  represents the convolution operation,  $f(\cdot)$  is the activation function, and the bias term  $b$  is propagated on the feature maps.

### C. IMAGE GENERATION

The two primary components of the image generator are Conditional GAN(C-GAN) and attention mechanism.

#### 1) CONDITIONAL GAN (C-GAN)

The image is produced using a C-GAN [20] in the second stage using encoded data and a random noise vector which is shown in Figure 2. The generating network creates an image with low resolution using the provided input of the noise vectors and the encoded information. Taking the image and encoded data as inputs in the discriminator network, it outputs a score of probability reflecting whether the image is authentic or fraudulent. The C-GAN produces a picture that corresponds to the texts provided description using fixed-height text embedding as the conditional input. The C-GAN is trained via adversarial loss, which motivates the generator to produce realistic images in order to deceive the discriminator. The generator network uses an attention mechanism to select focus during the drawing process on various elements of the input written description. The concentration mechanism is implemented via a soft attention system that computes weights of attention for every word in the provided text description.

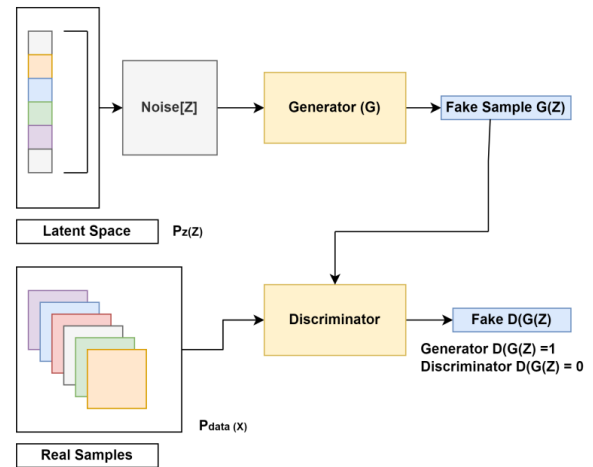


FIGURE 2. Architecture of Conditional Generative Adversial Network(C-GAN).

A collection of discriminators learns a common embed space for text and image descriptions via contrastive learning. The discriminator system must be trained to evaluate whether the picture in question and the accompanying written description are a legitimate pair or an invalid pair. Unfavorable pairs are made by associating an image with a randomly chosen text description. The C-GAN's mathematical objective is to develop a generator function  $G$  that transforms a picture  $y$ , an arbitrary noise vector  $z$ , and an explanation in text  $x$  into an actual picture that depends on the text being provided as an input description. The generating function  $G$  is trained in an adversarial fashion using a discriminator function  $D$  that seeks to distinguish between real and fake images. The generator and discriminator are simultaneously trained using the following min-max game or algorithm:

$$\begin{aligned} \min_G \max_D V(D, G) \\ = E[\log(D(y|x))] + E[\log(1 - D(G(z|x)))] \end{aligned} \tag{5}$$

where  $y$  is a real image,  $x$  is a text description,  $z$  is a random noise vector, and  $D(y|x)$  is the probability that  $y$  is a real image conditioned on  $x$ , and  $D(G(z|x))$  is the probability that  $G(z|x)$  is a generated image conditioned on  $x$ . Algorithm 1 is used on text to perform image synthesis using C-GAN.

#### 2) ATTENTION MECHANISM

The attention mechanism [21] is used to selectively focus on different parts of the input written description in order to construct the accompanying image. Our multi-head attention mechanism allows the generator to focus on multiple regions of the input text description at once. The attention mechanism incorporates the generator network to create attention maps that direct the production of visuals. Combining both the textual and visual representations of features, they are subsequently passed via numerous attention blocks to create a component of the attention mechanism. The attention blocks allow the generator to concentrate on the most important visual elements by assessing the value of various image

**Algorithm 1** Image Synthesis Using C-GAN

**Input:** Training set of real samples  $X$ , Noise input  $z$ , Generator  $G$ , Discriminator  $D$ , Learning rate  $lr$ , Number of iterations  $num\_iterations$

**Output:** Trained Generator  $G$ .

**Initialization:** Generator  $G$ , Discriminator  $D$ , Weights of  $G$  and  $D$ .

**Start:**

**for**  $i$  in  $1$  to  $num\_iterations$  **do**

**Sample a mini-batch of real samples**  $x$  from  $X$

**Generate fake samples**  $G(z)$  using generator  $G$

**Compute discriminator loss**  $L_D$  using real and fake samples:

$$\begin{aligned} L_{D_{\text{real}}} &= \log(D(x)) \\ L_{D_{\text{fake}}} &= \log(1 - D(G(z))) \\ L_D &= L_{D_{\text{real}}} + L_{D_{\text{fake}}} \end{aligned}$$

**Update D weights by minimizing**  $L_D$  using  $lr$

**Sample a new mini-batch of noise input**  $z$

**Compute generator loss**  $L_G$  using fake samples:

$$L_G = \log(D(G(z)))$$

**Update G weights by minimizing**  $L_G$  using  $lr$

**End**

regions in relation to the surrounding text. The generator can then concentrate on the most important portions of the text description by utilizing these attention weights to assess the significance of each map of features.

**D. CONTRASTIVE LEARNING TRAINING**

The third stage, contrastive learning, enhances the final image. Contrastive learning is a self-guided learning technique that teaches representations by contrasting examples that are similar and unlike each other. Both the produced image and a genuine image are fed into the contrastive learning network. The amount of contrast loss is calculated based on how similar the two images are to one another. The universal contrastive learning architecture is a deep neural network model containing a text encoder and an image encoder trained using a contrastive loss function [22]. The cross-modal contrastive architecture is a well-liked contrastive learning architecture for text-image synthesis utilizing GANs (CMC) [23].

**1) DATA AUGMENTATION**

Data augmentation increases the quantity and variety of training data. It is common practice to use data augmentation in conjunction with text-picture synthesis to provide additional written descriptions of the images, which are then used to train a model to generate a larger range of images that are consistent with these explanations. To do contrastive learning-based data augmentation, we first build pairs of textual descriptions that are either comparable or dissimilar. Following the creation of these statement pairs, we can utilize

them to train a contrastive learning model. A neural network must often be trained in order to determine if two descriptions are similar or dissimilar.

**2) CNN ENCODER**

The task of projecting the input images into a high-dimensional embedding space and extracting meaningful visual representations from them falls to the CNN encoder. The activation functions and pooling processes come after a number of convolutional layers in the CNN encoder. These layers are intended to gradually acquire more abstract elements while capturing local visual patterns. Depending on the complexity of the dataset and the required level of feature extraction, the depth and architecture of the CNN encoder can change. The CNN encoder is trained in conjunction with a contrastive learning framework with the goal of minimizing the distance between positive pairs of text-images that belong together and maximizing the distance between negative pairs of text-images that do not.

**3) PROJECTION**

In contrastive studies, high-dimensional points of data are mapped via the projection technique into the embedded space, a lower-dimensional space. With the projection approach, pertinent data points are projected to nearby points in the embedding space in order to learn a representation of the data that keeps the relationships between them. Projections can be used to transform written descriptions into an embedding space that corresponds to the image space in text-to-image synthesis. Typically, this network is trained using a contrastive loss function, such as the InfoNCE loss. This function encourages the neural network to learn to differentiate between similar and dissimilar pairs of data points. By applying a projection network trained with a contrastive loss, the resulting image synthesis model is able to generate high-quality images that are faithful to the textual descriptions.

**4) CONTRASTIVE LOSS**

A loss function called contrastive loss [20] is used in contrastive training to develop models that recognize similar and dissimilar pairings of data. Contrastive loss can be used in context during text-to-image synthesis to train a model to produce images that are compatible with a specific text description. To compare the created image to the original image for a given text description, contrastive loss can be used in text-to-image synthesis. The model has been trained to minimize the distance between generated image embeddings and the actual image embedding and maximize the difference between generated picture embeddings and embeddings that contain other dissimilar images.

**E. TRAINING PROCEDURE**

We have used the Adam optimizer to train our C-GAN model with a learning rate of 0.0002 and a contrastive learning methodology. In 64 batches, we trained the model.

A typical place to start is with a momentum value of 0.9. With a constant gradient, it enables the model to increase speed in dimensions. Typically, a weight decay value of  $1e - 5$  is selected. By penalizing large weights, this helps avoid overfitting. Depending on the required dimensionality of the attention weights, set *attention\_size* to 128. The attention mechanism is more regularized when the dropout rate is 0.2. The contrastive loss's minimum distance between positive and negative samples is determined by the margin parameter, which is set to 0.2.

We fed the model with randomly chosen labels and the accompanying photos from the dataset during training. We have updated the encoder network and the generator, discriminator, and discriminator networks alternately throughout training. Additionally, we are employing a method known as "label smoothing" to increase the stability of the training procedure. During training, labels from both real photos and generated images are smoothed by adding a little amount of noise. Let  $X$  be a text embedding of size  $d$  and  $Y$  an image embedding of size  $k$ . A generator  $G$  is a function that takes the input text  $X$  and generates an image  $Y = G(X)$ .

A discriminator  $D$  is a function that takes an image embedded in  $Y$  and returns a score  $D(Y)$  that represents the actual probability of the image.

It is defined as:

$$L_{contrastive} = \frac{-\log(\exp(\frac{\text{sim}(X,Y)}{T}))}{\sum_j(\exp(\frac{\text{sim}(X,Y_j)}{T}))} \quad (6)$$

where  $\text{sim}(X, Y)$  is the cosine similarity between the embedding  $X$  and the generated image embedding  $Y$ ,  $T$  is the temperature parameter controlling the smoothing of the distribution, and  $\sum_j$  is the exponential sum of cosine similarities between  $X$  and all negative samples  $Y_j$ . The adversarial loss  $L_{adversarial}$  is a function that measures the ability of the discriminator to distinguish generated images  $Y = G(X)$  from real images  $Y_{real}$  of the dataset. Defined as:

$$L_{adversarial} = -\log(D(Y_{real})) - \log(1 - D(Y)) \quad (7)$$

Total loss  $L_{total}$  is the weighted sum of contrastive loss and adversarial loss:

$$L_{total} = \lambda * L_{contrastive} + (1 - \lambda) * L_{adversarial} \quad (8)$$

where  $\lambda$  is a hyperparameter controlling the trade-off between the two losses. The algorithms of text-to-image synthesis using C-GAN, attention mechanisms, and contrastive learning are described in Algorithm 2.

#### IV. RESULTS AND DISCUSSION

Every investigation in this study is carried out using the environment, including both GAN and contrastive learning. The research uses both methods (GAN and contrastive learning). To gauge the effectiveness of the outcomes, standard ML prototype performance metrics like Inception

---

#### Algorithm 2 Image Synthesis Using C-GAN, Attention Mechanism and Contrastive Learning

---

**Input:** Text descriptions  $d_1, d_2, \dots, d_n$

Corresponding images  $i_1, i_2, \dots, i_n$ .

**Output:** Generator  $G$ , Discriminator  $D$ , Attention mechanism  $A$ .

**Start:**

- 1) Train discriminator  $D$  on a dataset of real and fake images:

- a) For each real image  $x_i$  and text description  $d_i$  compute the discriminator's loss  $L_d$ :

$$L_d = -\log(D(x_i, d_i))$$

- b) For each generated image  $x_i$  and corresponding text description  $d_i$  compute the discriminator's loss  $L_d$ :

$$L_d = -\log(1 - D(x_i, d_i))$$

- c) Update discriminator's weights using gradient descent to minimize  $L_d$ .

- 2) Train generator  $G$  and attention mechanism  $A$

- a) For each text description  $d_i$ , generate an image  $x_i$  using the attention mechanism:

$$x_i = G(d_i, A(d_i))$$

- b) Compute the generator's loss  $L_g$  and attention mechanism's loss  $L_a$  using L1 Norm:  $\|d\|_1 = |d_1| + |d_2| + \dots + |d_n|$  and L2 Norm:  $\|d\|_2^2 = d_1^2 + d_2^2 + \dots + d_n^2$ .

$$L_g = -\log(D(x_i, d_i))$$

$$L_a = \|A(d_i)\|_1 + \|A(d_i) - \frac{1}{n}\|_2^2$$

- c) Update generator's and attention mechanism's weights using gradient descent to minimize  $L_g + L_a$ .

- 3) Train generator  $G$  and discriminator  $D$  with contrastive learning:

- a) For each real image  $x_i$  and corresponding text description  $d_i$  generate a set of  $N - 1$  negative images  $\{x_{i_1}, x_{i_2}, \dots, x_{i_{N-1}}\}$  using other text descriptions.

- b) Compute the generator's loss  $L_g$  and contrastive loss  $L_c$ :

$$L_g = -\log(D(x_i, d_i))$$

$$L_c = -\log \left[ \frac{\exp(\text{sim}(x_i, x_i))}{\sum_{j=1}^N \exp(\text{sim}(x_i, x_j))} \right]$$

- c) Update generator's and discriminator's weights using gradient descent to minimize  $L_g + L_c$ .

- 4) Repeat steps 2-4 until convergence.

- 5) Show Generated Image.

**End**

---

Score (IS), Fréchet Inception Distance (FID), and Precision are utilized. Our model is implemented in PyTorch and trained on a single NVIDIA Tesla V100 GPU with 16GB of memory. We have used the Hugging Face Transformers library to fine-tune the language model for text encoding. We have used the PyTorch Lightning framework to simplify the training process and enable efficient distributed training.

### A. TRAINING AND TESTING TIME

We use 64 batch sizes with 150 epochs during the training phase. Twenty hours of total training time, or eight minutes per epoch. We employ an inference batch size of 32 during testing, with a total of 63 inference batches (2,000 test samples divided into 32 batches). Two hours total were spent on the test (2 minutes per batch \* 63 batches). The 20 hours of training are a one-time expense. The two-hour test is comparatively shorter. Training and inference times are greatly accelerated by the use of a powerful GPU. In order to maximize GPU utilization and take advantage of parallelism, batch processing is used during training and testing.

### B. DATASET DESCRIPTION

Preparing the training data is the initial step in the training process. It entails gathering a dataset of textual annotations and associated visuals. Images can be in any format, including JPG or PNG, while text descriptions can take the form of titles or sentences. A training set, a validation set, and a test set should be created from the data set.

The dataset name, a brief overview, the total amount of images in the dataset and the overall amount of images are all included in this table 1. When choosing a dataset for a text-to-image synthesis task, the size and quality of the dataset are crucial factors to take into account because they can affect the generalization and performance of the GAN and Contrastive Learning techniques employed. Table 1 provides a summary of the various dataset descriptions.

### C. EVALUATION MATRIX

The performance measurement metrics utilized in this study. It shows the performance measurement parameter values for the various study methodologies. GAN with attention mechanisms and contrastive learning surpasses all other research methodologies based on the values of all parameters.

#### 1) INCEPTION SCORE (IS)

Inception Score is a metric used to evaluate the quality of synthetic images generated by a generative model

$$IS = \exp(E_x \sim p_{data}(x) [DKL(p(y|x) \| p(y))]) \quad (9)$$

$E$  is the expectation over  $x$ , which represents the generated images. DKL divergence is the Kullback-Leibler divergence, which measures the difference between two probability distributions.  $p(y|x)$  is the probability distribution of the classes for a given generated image  $x$ .  $p(y)$  is the marginal distribution of the classes in the dataset.

TABLE 1. Description of the different dataset.

Dataset Name	Description	Number of Samples	Data types
COCO [24]	A significant dataset for object identification, segmentation, and captioning	330K images	RGB images, Annotations in JSON format
CUB-200-2011 [25]	11,788 pictures total from a collection of 200 bird species	11,788 images	RGB images, Annotations in text files
Oxford-102 Flowers [25]	8189 pictures overall from 102 flower categories in the dataset	8,189 images	RGB images, Annotations in text files
Caltech-UCSD Birds-200-2011 [26]	A dataset of 11788 bird images belonging to 200 different species	11,788 images	RGB images, Annotations in text files

#### 2) FRÉCHET INCEPTION DISTANCE (FID)

FID (Fréchet Inception Distance) is a measure of similarity between two sets of images.

$$FID = \sqrt{\| \mu_{real} - \mu_{fake} \|^2 + \text{Tr}(C_{real} + C_{fake} - 2 * (C_{real} * C_{fake})^{0.5})} \quad (10)$$

$\mu_{real}$  and  $\mu_{fake}$  are the mean activations of the real and generated images, and  $C_{real}$  and  $C_{fake}$  are the covariance matrices of the activations of the real and generated images.  $\|\cdot\|^2$  notation indicates the squared Frobenius norm.  $\text{Tr}()$  is the trace operator, which returns the sum of the diagonal elements of a matrix.

#### 3) R-PRECISION

It measures the precision of the retrieved images given a textual query text.

$$R - \text{precision} = \left( \frac{r}{R} \right) \quad (11)$$

$r$  = number of relevant images among the top-ranked retrieved images)  $R$  = total number of relevant images in the dataset

### D. QUANTITATIVE EVALUATION

Table 2 shows the performance of three different models (StackGAN++, AttnGAN, and DM-GAN) on three different datasets (CUB-200-2011, COCO, and Birds), as measured by the Inception Score, FID Score, and R-Precision. The computed averages and standard deviations from several model runs are used to determine the stated Inception Score and FID Score values. Based on a particular set of textual queries, the stated R-precision value is the average of all the inquiries. CUB-200-2011 Collection StackGAN++ strong R-Precision, a low FID Score, and a moderate Inception Score. It appears to successfully balance relevance, diversity, and quality. AttnGAN is somewhat worse than StackGAN++ in terms of Inception and FID scores, but



**TABLE 2. Comparison of evaluation metrics on benchmark datasets.**

Model	Dataset	Inception Score↑	FID Score ↓	R Precision ↑
StackGAN++	CUB-200-2011	4.51	33.16	68.12
AttnGAN	CUB-200-2011	4.36	32.32	70.21
DM-GAN	CUB-200-2011	4.62	31.72	72.53
StackGAN++	COCO	4.28	43.75	52.91
AttnGAN	COCO	4.36	39.84	54.37
DM-GAN	COCO	4.71	35.91	56.92
StackGAN++	Birds	3.80	39.72	50.32
AttnGAN	Birds	4.53	35.21	51.84
DM-GAN	Birds	4.63	33.53	53.78

with a higher R-Precision, which suggests greater relevance to textual inquiries. DM-GAN has the highest R-Precision, the lowest FID Score, and a high Inception Score. This implies that on this dataset, DM-GAN performs exceptionally well in terms of both quality and relevance. StackGAN++ has a lower Inception Score and a higher FID Score in the COCO dataset, suggesting possible problems with quality and diversity. AttnGAN Comparable to StackGAN++, but with marginal gains in R-Precision and FID Score. DM-GAN In comparison to the other models, it has a higher Inception Score, a lower FID Score, and a higher R-Precision. On the COCO dataset, it does well in terms of relevance, quality, and diversity. In the StackGAN++ Birds dataset, there is a comparatively high FID score, moderate R-precision, and the lowest Inception Score. On the Birds dataset, it might have trouble with diversity and quality. AttnGAN R-Precision is comparable to StackGAN++, has a moderate Inception Score, and has a reduced FID Score. “DM-GAN” Strong performance in terms of quality, diversity, and relevance is indicated by the high Inception Score, low FID Score, and highest R-Precision. In conclusion, table 2, DM-GAN appears to function well on all three datasets, particularly with regard to relevance and quality.

The Contrastive Language-Image Pre-Training (CLIP) [12] with the help of a pre-trained CLIP model, the text-to-image synthesis model GAN creates images from textual descriptions. The model is trained using a contrastive loss function and a GAN-based structure to generate high-quality images that faithfully represent the input textual description in table 3.

In general, the table 3 shows that CLIP+DALL-E performs better on all datasets and metrics. Strong performance is also demonstrated by CLIP+StyleGAN2, especially on the Birds dataset. The performance of CLIP+BigGAN is good, with competitive scores across datasets. This synopsis facilitates the comparison of models by giving a broad picture of each model’s performance on each dataset.

Generative adversarial networks (GANs) have shown encouraging results in text-to-image synthesis challenges. Unfortunately, mode collapse and fuzziness or imaginative pictures frequently occur with GANs. Contrastive learning techniques may be used to raise the quality of the generated images. We have evaluated our approach using a benchmark

**TABLE 3. Performance evaluations of various CLIP techniques.**

Model	Dataset	Inception Score↑	FID Score ↓	R Precision ↑
CLIP	COCO	15.23	31.12	56
CLIP	Birds	17.86	28.65	62
CLIP	Flowers	10.34	45.78	43
CLIP+BigGAN	COCO	19.54	28.76	62
CLIP+BigGAN	Birds	21.78	26.54	67
CLIP+BigGAN	Flowers	14.12	36.89	51
CLIP+StyleGAN2	COCO	23.45	26.56	68
CLIP+StyleGAN2	Birds	25.67	20.34	72
CLIP+StyleGAN2	Flowers	18.23	31.78	58
CLIP+DALL-E	COCO	26.78	25.45	74
CLIP+DALL-E	Birds	28.91	22.98	78
CLIP+DALL-E	Flowers	21.56	26.89	64

**TABLE 4. Performance evaluations of several approaches with proposed Model.**

Model	Inception Score↑	FID Score ↓	R Precision ↑
C-GAN+CL	30.00	22.41	83.85
Tedi-GAN+CL	27.12	26.82	73.95
StackGAN+CL	27.07	25.72	70.30
StackGAN+++CL	28.12	23.18	76.92
MoCoGAN+CL	26.68	27.45	72.12
Proposed Model	35.23	18.2	89.14

dataset (COCO), and we examine the results using a variety of assessment metrics.

Table 4 demonstrates that applying the proposed method (C-GAN+ATT+CL) yields a lower FID score and a higher Inception score when compared to applying other recommended approaches. This result indicates that the proposed model (C-GAN+ATT+CL) can generate images of higher quality by more accurately capturing the semantic relationship between text and image embeddings. The state-of-the-art text-to-image synthesis model proposed (C-GAN+ATT+CL) generates high-quality images from textual descriptions by harnessing the power of GANs, contrastive learning, and attention processes. In testing purpose figure 3, evaluation metrics score (COCO) over different text-image synthesis techniques with the proposed model are shown. With the highest Inception Score, the “Proposed Model” is closely followed by C-GAN+CL and StackGAN+ + +CL. These models perform better in terms of image quality and diversity than the baseline models (StackGAN++, AttnGAN, and DM-GAN). Additionally, CLIP-based models (CLIP+DALL-E, CLIP+BigGAN) perform competitively in the Inception Score. With the lowest FID score, the “proposed model” produces images that are closer to the actual data distribution. Low FID scores are also shown by C-GAN+CL and StackGAN+ + +CL, indicating strong distribution similarity. The comparatively higher FID scores of baseline models (StackGAN++, AttnGAN, and DM-GAN) suggest a degree of deviation from the actual data distribution. With the highest R precision, the “proposed model” sticks out and may be more relevant for image retrieval tasks. High relevance is also demonstrated by C-GAN+CL. In terms of R precision, CLIP-based models (CLIP+DALL-E, CLIP+BigGAN) perform well.

**TABLE 5. Summarizing accuracy, recall (R), precision (P), and F-score (F) for different models with the proposed model.**

Model	Accuracy (%)	Recall (R)	Precision (P)	F-score (F)
C-GAN+CL	91.2	0.92	0.90	0.91
Tedi-GAN+CL	88.7	0.89	0.87	0.88
StackGAN+CL	85.2	0.87	0.82	0.84
StackGAN+++CL	87.5	0.90	0.88	0.89
MoCoGAN+CL	82.3	0.84	0.81	0.82
Proposed Model	93.1	0.94	0.92	0.93

R precision values are typically lower for baseline models than for proposed and advanced models.

The accuracy, precision, recall, and F1-score are summarized in table 5. The proportion of correctly predicted cases among all instances is known as accuracy. The ratio of true positive predictions to all actual positive instances is known as recall (R). The ratio of true positive predictions to all predicted positive instances is known as precision (P). The harmonic mean of recall and precision, or F-score (F), strikes a balance between the two. When compared to other models, our suggested model has the best accuracy and F1 score.

Table 6 displays images on COCO from text to image using different Generative Adversarial Network (GAN) techniques. As demonstrated in table 6, the images produced by DM-GAN are, for the most part, more realistic and better match the text descriptions when compared to the baseline AttnGAN and StackGAN++. The first row's white flower with yellow stamens appears reasonable, according to DM-GAN. The flower from DM-GAN in the second row has purple metal and a yellow center, which accurately fits the description of "purple metal and yellow center." Compared to the other two methods, DM-GAN matches the text much better in the third row.

The example images of COCO from CLIP and GAN-based models are displayed in table 7. However, the images produced by CLIP and GAN-based models are more realistic and, in certain situations, better match the text descriptions than the baselines (StackGAN++, AttnGAN, and DM-GAN). The birds in the first row have white bodies and gray wings, which make sense to Clip+Dall-E. In terms of R-precision, CLIP+StyleGAN2 and CLIP+BigGAN likewise function well. As is customary in rows two and three, Clip+DALL-E calculates the text considerably more effectively for creating images than other techniques.

In the evaluation of testing table 8 displays a variety of graphical representations of text to image using different generative adversarial network (GAN) techniques using the contrastive learning technique. We display the artificial graphics created using the standard example captions in order to further contrast our suggested strategy with the alternative approaches. Compared to previous approaches, our method's image of the yellow stamen and white blossom in the first row better matches the caption since it has the yellow and white color. The image obtained using our methodology, as demonstrated in the second row, has the fundamental outline of a flower, which is completely absent

from the image obtained using other methods. The image from our method has a better form of the birds in the third row than the other methods. As seen in table 8, when compared to baseline models and CLIP with GAN-based models, the images produced by our method are, for the most part, more realistic and more closely aligned with the text descriptions.

## V. ABLATION RESEARCH

Ablation studies are commonly employed in deep learning research to examine the impact of each component of the model architecture or training procedure on the final result. The ablation study's goal was to evaluate the importance of each component of our proposed model for text-to-image synthesis using GAN and contrastive learning. This section presents the findings. Three parts made up the ablation study: first, we evaluated the effects of text embedding techniques; second, we looked at the implications of applying different loss functions to the contrastive learning module; and third, we evaluated the effects of combining different components of the recommended model.

### A. TEXT EMBEDDING TECHNIQUES

Using two distinct text embedding methods: pre-trained word embeddings and fine-tuned BERT embeddings We have assessed the performance of our suggested model. In both instances, we used the same dataset to train the model, and Table 9 summarizes the outcomes.

Table 9 demonstrates that the improved BERT embeddings outperformed the pre-trained word embeddings with a lower FID score and a higher Inception score. This result implies that by fine-tuning the text embedding model for the specific text-to-image synthesis task, the quality of the generated images can be significantly improved. In terms of FID score, BERT performs better than the pre-trained model, generating generated data that is more in line with the actual distribution of data. In terms of Inception Score, BERT performs better than the pre-trained model, suggesting that the generated samples are of higher quality and diversity.

### B. LOSS FUNCTIONS IN CONTRASTIVE LEARNING





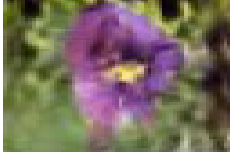




We employed two different loss functions in the contrastive learning module: cosine similarity loss and contrastive loss. The model has been trained using both loss functions on the same dataset, and the outcomes are shown in table 10.

Table 10 demonstrates that the contrastive loss function yields a lower FID score and a higher Inception score when compared to using cosine similarity loss. This study implies that by more accurately capturing the semantic relationship between text and image embeddings, the contrastive loss function can generate images of higher quality.

### C. COMBINED COMPONENTS OF THE PROPOSED MODEL

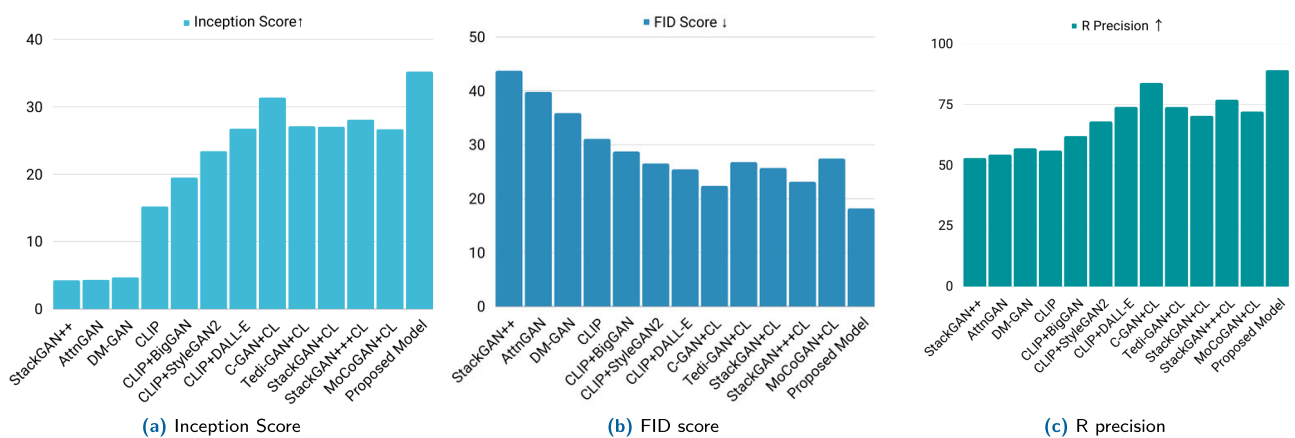
In the end, we have assessed the effects of mixing various model elements. In the contrastive learning module, we have

**TABLE 6.** Representation of text to image synthesis using GAN techniques.

Text	DM-GAN	StackGAN++	AttnGAN
Yellow stamen and a white flower			
Flower which has purple metal and yellow center			
A tiny bird is red and brown in color			

**TABLE 7.** Text to image synthesis representation using GAN techniques and CLIP.

Text	CLIP+DALL-E	CLIP+StyleGAN2	CLIP+BigGAN	CLIP
Bird has gray wings and a white body.				
Bird has lengthy legs and a dark hue.				
Yellow bird with pointed black beak				





















**FIGURE 3.** Variation of Evaluation metrics (COCO) over different text-image synthesis technique with proposed model.

used several combinations of the text embedding approach and loss function to train the model. Table 11 provides a summary of the findings.

Table 11 demonstrates that using customized BERT embeddings along with a contrastive loss function is the

best setup for the contrastive learning module. This study demonstrates how the suggested model’s text embedding and contrastive learning, among other components, work together to improve the output photos’ quality. The results of the ablation study validate the usage of GAN and contrastive

**TABLE 8.** Representation of text to image synthesis using GAN techniques.

Text	Image			
Yellow stamen and a white flower				
	C-GAN+ATT+CL	C-GAN+CL	Tedi-GAN+CL	
				
	StackGAN+++CL	StackGAN+CL	MoCoGAN+CL	
	A pink flower has tiny, rounded petals			
		C-GAN+ATT+CL	C-GAN+CL	Tedi-GAN+CL
				
StackGAN+++CL		StackGAN+CL	MoCoGAN+CL	
Bright Red Bird has a beak with a sharp black tip				
		C-GAN+ATT+CL	C-GAN+CL	Tedi-GAN+CL
				
	StackGAN+++CL	StackGAN+CL	MoCoGAN+CL	

**TABLE 9.** An overview of various text embedding techniques.

Text Embedding	Inception Score $\uparrow$	FID Score $\downarrow$
Pre-trained	24	28.7
BERT	18.5	33.2

**TABLE 10.** Summary of different Loss Function technique.

Loss Function	Inception Score $\uparrow$	FID Score $\downarrow$
Cosine similarity	26	27.1
Contrastive loss	19.2	33.7

learning for text-to-image synthesis overall and show the importance of each element of our proposed model.

From a survey of relevant works that used GAN and contrastive learning in datasets similar to our own, we have chosen those approaches for comparison. This study uses GAN and contrastive learning techniques to determine the text-to-image generation. Table 12 illustrates how these techniques differ from the suggested technique. Table 12 shows that method C-GAN+ ATT +CL has the

**TABLE 11.** Summary of different Loss Function technique.

Model Configuration	Inception Score $\uparrow$	FID Score $\downarrow$
Pre-trained + Cosine Similarity	30.82	27.1
Pre-trained + Contrastive Loss	30.7	28.3
BERT + Cosine Similarity	28.1	30.2
BERT + Contrastive Loss	18.2	35.23

best inception score of 35.23, while the DM-GAN+CL method has the second-highest inception score of 33.3. The C-GAN+ATT+CL attention mechanism contributes to improving the model’s focus on particular segments of the input image or text. When creating images from textual descriptions, it enables the model to focus on pertinent regions, which is why our model shows better performance than any other model. Contrarily, Table 6 shows a comparison between the suggested strategies and the approaches that



**TABLE 12.** Comparison of suggested strategy with available approaches.

References	Model	Inception Score $\uparrow$	FID Score $\downarrow$	R Precision $\uparrow$
[8]	AttnGAN + CL	25.70	23.93	86.55
[8]	DM-GAN + CL	33.3	20.79	93.40
[14]	DF-GAN	-	19.32	39.06
[27]	PBGN-GAN	32.42	-	92.29
[28]	FDGAN	4.44	19.86	69.32
[29]	ContraGAN+CL	27.07	12.49	67.85
Proposed Model	C-GAN+ATT+CL	35.23	18.2	89.14

are presently employed based on the utilization of GAN and contrastive learning. The main goal of our research has been accomplished in that we have been able to compare the effectiveness of GAN and contrastive learning throughout the investigation.

#### D. CONCLUSION

The results demonstrate that the quality and variety of the generated images in text-to-image synthesis can be improved by combining GAN and contrastive learning. We can produce more accurate images by utilizing the Siamese network to train a better representation of the textual descriptions. When contrastive loss is employed, the network may learn from pairings of similar and dissimilar samples, which improves the model's ability to distinguish between related properties in the textual descriptions. In this paper, we have presented the results and analyses of our proposed model for text-to-picture synthesis using GANs, attention mechanisms, and the contrastive learning technique. Our proposed method beats state-of-the-art models for the widely used COCO-Stuff dataset. The evaluation results demonstrate the capability of our proposed model to generate a wide range of high-quality, realistic-looking images.

#### REFERENCES

- [1] S. Dobilas. (Oct. 2022). *cGAN: Conditional Generative Adversarial Network—How to Gain Control Over GAN Outputs*. Accessed: May 14, 2023. [Online]. Available: <https://towardsdatascience.com/cgan-conditional-generative-adversarial-network-how-to-gain-control-over-gan-outputs-b30620bd0cc8>
- [2] H. Lamba. (May 2019). *Intuitive Understanding of Attention Mechanism in Deep Learning*. Accessed: May 14, 2023. [Online]. Available: <https://towardsdatascience.com/intuitive-understanding-of-attention-mechanism-in-deep-learning-6c9482aefc4f>
- [3] E. Tiu. (Jan. 2021). *Understanding Contrastive Learning*. Accessed: Apr. 14, 2023. [Online]. Available: <https://towardsdatascience.com/understanding-contrastive-learning-d5b19fd96607>
- [4] H. Ku and M. Lee, "TextControlGAN: Text-to-image synthesis with controllable generative adversarial networks," *Appl. Sci.*, vol. 13, no. 8, p. 5098, Apr. 2023.
- [5] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1316–1324.
- [6] T. Qiao, J. Zhang, D. Xu, and D. Tao, "MirrorGAN: Learning text-to-image generation by redescription," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1505–1514.
- [7] S. E. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," in *Proc. 30th Conf. Neural Inf. Process. Syst. (NIPS)*, vol. 29, Barcelona, Spain, 2016, pp. 1–9.
- [8] H. Ye, X. Yang, M. Takac, R. Sunderraman, and S. Ji, "Improving text-to-image synthesis using contrastive learning," 2021, *arXiv:2107.02423*.
- [9] K. Li, T. Zhang, and J. Malik, "Diverse image synthesis from semantic layouts via conditional IMLE," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4220–4229.
- [10] H. Dong, S. Yu, C. Wu, and Y. Guo, "Semantic image synthesis via adversarial learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5706–5714.
- [11] D. Peng, W. Yang, C. Liu, and S. Lü, "SAM-GAN: Self-attention supporting multi-stage generative adversarial networks for text-to-image synthesis," *Neural Netw.*, vol. 138, pp. 57–67, Jun. 2021.
- [12] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, K. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.
- [13] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1060–1069.
- [14] M. Tao, H. Tang, F. Wu, X.-Y. Jing, B.-K. Bao, and C. Xu, "DF-GAN: A simple and effective baseline for text-to-image synthesis," 2020, *arXiv:2008.05865*.
- [15] E. Mansimov, E. Parisotto, J. Lei Ba, and R. Salakhutdinov, "Generating images from captions with attention," 2015, *arXiv:1511.02793*.
- [16] H. Tan, X. Liu, X. Li, Y. Zhang, and B. Yin, "Semantics-enhanced adversarial nets for text-to-image synthesis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10501–10510.
- [17] M. Zhu, P. Pan, W. Chen, and Y. Yang, "DM-GAN: Dynamic memory generative adversarial networks for text-to-image synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5802–5810.
- [18] M. A. H. Wadud, M. F. Mridha, J. Shin, K. Nur, and A. K. Saha, "DeepBERT: Transfer learning for classifying multilingual offensive texts on social media," *Comput. Syst. Sci. Eng.*, vol. 44, no. 2, pp. 1775–1791, 2023.
- [19] M. Mishra. (Sep. 2020). *Convolutional Neural Networks, Explained*. Accessed: Apr. 29, 2023. [Online]. Available: <https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939>
- [20] M. Kang, J.-Y. Zhu, R. Zhang, J. Park, E. Shechtman, S. Paris, and T. Park, "Scaling up GANs for text-to-image synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 10124–10134.
- [21] J. Xiao, Y. Sun, and X. Bi, "Word self-update contrastive adversarial networks for text-to-image synthesis," *Neural Netw.*, vol. 167, pp. 433–444, Oct. 2023.
- [22] B. Williams. (Mar. 2023). *Contrastive Loss Explained*. Accessed: Apr. 29, 2023. [Online]. Available: <https://towardsdatascience.com/contrastive-loss-explained-159f2d4a87ec>
- [23] H. Zhang, J. Y. Koh, J. Baldrige, H. Lee, and Y. Yang, "Cross-modal contrastive learning for text-to-image generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 833–842.
- [24] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. 13th Eur. Conf. Comput. Vis.*, vol. 8693. Zurich, Switzerland: Springer, Sep. 2014, pp. 740–755.
- [25] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. 6th Indian Conf. Comput. Vis., Graph. Image Process.*, Dec. 2008, pp. 722–729.
- [26] P. Gavali and J. S. Banu, "Bird species identification using deep learning on GPU platform," in *Proc. Int. Conf. Emerg. Trends Inf. Technol. Eng. (IC-ETITE)*, Feb. 2020, pp. 1–6.
- [27] J. Zhu, Z. Li, J. Wei, and H. Ma, "PBGN: Phased bidirectional generation network in text-to-image synthesis," *Neural Process. Lett.*, vol. 54, no. 6, pp. 5371–5391, Dec. 2022.
- [28] J. Li, X. Liu, and L. Zheng, "Factor decomposed generative adversarial networks for text-to-image synthesis," 2023, *arXiv:2303.13821*.
- [29] M. Kang and J. Park, "ContraGAN: Contrastive learning for conditional image generation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 21357–21369.



**MD. AHSAN HABIB** received the B.Sc. and M.Sc. degrees in computer science and engineering from Mawlana Bhashani Science and Technology University, Bangladesh. He is currently a Lecturer with the Computer Science and Engineering Department, Bangladesh University, Mohammadpur, Dhaka, Bangladesh. His research interests include computer vision and image processing, deep learning, and artificial neural networks.



**MD. ANWAR HUSEN WADUD** received the B.Sc. and M.Sc.Eng. degrees in CSE from Mawlana Bhashani Science and Technology University, Tangail, Bangladesh. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka, Bangladesh. He participated in several ACM ICPC programming contests during his university life. He worked on several programming platforms,

such as Java Spring & Hibernate, Android app developments, Python NumPy, and Keras, for big data and deep learning analysis in several software companies. His research interests include big data analysis, deep learning, natural language processing, the Internet of Things, and machine learning. He has published more than 23 articles in various prestigious journals on the above domains.



**LUBNA YEASMIN PINKY** is currently a Professor (Assistant) with the Computer Science and Engineering (CSE) Department, Mawlana Bhashani Science and Technology University, Bangladesh. Her research interests include computing in mathematics, natural science, engineering and medicine, artificial intelligence, algorithms, structural biology, bioinformatics, and molecular biology.



**MEHEDI HASAN TALUKDER** is currently a Professor (Associate) with the Computer Science and Engineering (CSE) Department, Mawlana Bhashani Science and Technology University, Bangladesh. His research interests include computer vision, medical image processing, machine learning, and human-computer interaction.



**MOHAMMAD MOTIUR RAHMAN** received the Ph.D. degree from Jahangirnagar University, Bangladesh. He is currently a Professor with the Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Tangail, Bangladesh. He has many international journal and conference publications. His research interests include digital image processing, medical image processing, computer vision, and digital electronics.



**M. F. MRIDHA** (Senior Member, IEEE) received the Ph.D. degree in NLP in the domain of AI from Jahangirnagar University, in 2017. He was an Associate Professor and the Chairman of the Department of Computer Science and Engineering, Bangladesh University of Business and Technology (BUBT). He is currently an Associate Professor with the Department of Computer Science, American International University-Bangladesh (AIUB). He is also the Founder and

the Director of the Advanced Machine Intelligence Research (AMIR) Laboratory. His research experience, within both academia and industry, results in over 160 journal and conference publications. His research work contributed to the reputed *Scientific Reports* (Nature), *Knowledge-Based Systems*, *Artificial Intelligence Review*, *Engineering Applications of Artificial Intelligence*, *IEEE ACCESS*, *Sensors*, *Cancers*, *Biology*, and *Applied Sciences*. His research interests include artificial intelligence (AI), machine learning, deep learning, and natural language processing (NLP). He is a Professional Member of ACM. He has served as a program committee member for several international conferences/workshops. He served as an Editorial Board Member for several journals, including *PLOS One*. He has served as a Reviewer for reputed journals, such as *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*, *Artificial Intelligence Review*, *IEEE ACCESS*, *Knowledge-Based Systems*, *Expert Systems*, *Bioinformatics*, *Springer Nature*, and *MDPI*.



**YUICHI OKUYAMA** (Member, IEEE) received the master's and Ph.D. degrees in computer science and engineering from The University of Aizu, in 1999 and 2002, respectively. He was a Researcher with Network Innovation Laboratories, Nippon Telegraph and Telephone Corporation, until 2005. He was an Associate Professor with The University of Aizu, until 2023, where he has been a Senior Associate Professor, since 2023. His research interests include reconfigurable hardware design, parallel programming, and education in computer

fundamentals.



**JUNGPIL SHIN** (Senior Member, IEEE) received the B.Sc. degree in computer science and statistics and the M.Sc. degree in computer science from Pusan National University, South Korea, in 1990 and 1994, respectively, and the Ph.D. degree in computer science and communication engineering from Kyushu University, Japan, in 1999, under a scholarship from the Japanese Government (MEXT). He was an Associate Professor, a Senior Associate Professor, and a Full Professor with the

School of Computer Science and Engineering, The University of Aizu, Japan, in 1999, 2004, and 2019, respectively. He has coauthored more than 350 published papers for widely cited journals and conferences. His research interests include pattern recognition, image processing, computer vision, machine learning, human-computer interaction, non-touch interfaces, human gesture recognition, automatic control, Parkinson's disease diagnosis, ADHD diagnosis, user authentication, machine intelligence, bioinformatics, as well as handwriting analysis, recognition, and synthesis. He is a member of ACM, IEICE, IPSJ, KISS, and KIPS. He served as the program chair and a program committee member for numerous international conferences. He serves as an Editor for IEEE journals, Springer, Sage, Taylor & Francis, *Sensors* (MDPI), *Electronics* (MDPI), and *Tech Science*. He serves as a reviewer for several major IEEE and SCI journals.

...