Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 218 (2023) 845–857

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Machine Learning and Data Engineering

Blended multi-class text to image synthesis GANs with RoBerTa and Mask R-CNN

Siddharth M[a], R Aarthi[b]

[a]Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, India
[b]Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, India

Abstract

Generation of scenes from text description requires employing computer vision for processing image and natural language processing for decoding the text provided. The task here is difficult as it requires generating multiple images of different classes together. Existing methods construct images from captions using a single dataset class, utilising Generative Adversarial Networks (GANs) approaches. Training several classes with GANs is a cumbersome task that needs a large amount of data and huge computational power. With complex datasets the efficiency of the generated image according to text description may be poor. In this paper, we propose an application that generates images based on the Caltech-UCSD bird and Oxford 102 flowers dataset. It leverages the Attentional Generative Adversarial Network (AttnGANs) as the generative model and the RoBERTa neural language model for word embeddings to build an image from multiple classes trained in isolation. The image created by GANs is segmented using Mask R-CNN and blended together using Poisson blending. The application can create scenes based on the description provided by the user and produce them in less time compared to training multiple classes in a single generative network.

Keywords: Computer Vision;Natural Language Processing; RoBERTa; Mask R-CNN; AttnGAN; Poisson Blending

1. Introduction

Image generation has been widely followed by various research in the past with generative networks and autoencoders. Text to image generation has developed recently with various applications in the industry for quick generation of images for artists, game designers, and filmmakers. This problem focuses on the use of both computer vision and natural language processing. The current research uses a single class of images for training and can only generate text

* Corresponding author. Tel.: +91-422-2685000 ; fax: +0-000-000-0000.
E-mail address: r_aarthi@cb.amrita.edu

1877-0509
10.1016/j.procs.2023.01.065

for that particular class at a given time. With multi-classes like the MS-COCO dataset [1], the training data tends to be very huge and complex. This requires a lot of time for the training and does not always guarantee the desired result. This paper comes up with a novel architecture that trains each class in isolation and makes use of Mask R-CNN for segmention of the images for its mask generation. Then, based on the information of each class provided by the user, the masks of each are blended to generate an image with multiple classes together in a single image. This aims to reduce the training size as each of the classes can now be trained individually and does not require a huge amount of mixed class data. This application can come in very handy for providing artists with a quick start idea for their pictures. Deep Convolutional GANs[2] are generally utilised as generative networks as they are used for image synthesis from description of provided text along with attention networks. Recent language models created have been shown to be quite effective for undestanding the text syntactically. In this paper we use transformers model in order to boost the problem's overall performance and elicit a better response from natural language. The RoBERTa model features an effective attention mechanism that can capture the meaning of words related to the class description [5]. The most recent state-of-the-art model uses single classes of data to read the description of a caption and learn the specifics such as location, shape, and posture of the class to link with the new description supplied and produce pictures based on these aspects. As additional classes are added, the scene becomes more complicated, and it becomes difficult to construct everything based on the supplied description and simultaneously update the background environment. The application requires the user to provide a text that has information related to both the classes and background details. For research purposes, birds and flowers are described in this paper. Birds' descriptions like "this big bird has a large, rounded green beak with blue feathers" and flower descriptions like "flower with long white petals and very long purple stamen" are provided to the system along with the positional details like how one class is placed over the other. The text description of each class is used in isolation to train the network for image to text generation. For the generative networks, the text to Image generative model with attention networks using RoBERTa pre-tained language model [6] is used. This model makes use of the attention networks and Deep Attentional Multimodal Similarity Model (DAMSM) deployed in AttnGANs. RoBERTa neural language model is employed to embed words and transform it to a particular feature representation. RoBERTa model makes use of attention mechanism to obtain crucial details assosiated with each words, and they can match these specefic connections in the attention heads. AttnGANs is utilised in the training of generative networks.in every successive stage, images of better resolution gets generated. With RoBERTa GANs, the Fréchet inception distance (FID) score is reduced to 20.77 from 23.98 in the main AttnGANs paper [7] for the CUB Birds dataset. So, once the training is completed, we generate the images based on the text description. The image generated is then used in Mask-RCNN to generate a mask of the segmented image. In this paper, we use the mask of a flower to place it in a particular position in the bird picture. Mask-RCNN [8] is used by training each custom dataset and then using the trained model for image segmentation. Furthermore, Poisson blending [12] is used for the blending of both images together in the position specified by the user. This approach also ensures that the colour of the inserted picture is altered, making the inserted item appear to be a part of the target image's surroundings. Therefore, the colour of a bright item that is copied and pasted into a relatively dark picture will become darker. This paper discusses about works ongoing in text to image area and the current scores each state-of-the-art models are getting. Then the datasets used for work have been discussed. The algorithm for the complete experimentation have been explained followed with all the methods involving the research. Finally, the results of the experiments and application developed are displayed and the future scope of project is discussed in the conclusion part. The primary advantage of conducting research on this issue is the development of an application that may assist artists in creating a preliminary sketch for their artwork. This can save a substantial amount of time in terms of business and finances. It will also help us investigate how the most recent language models that include attention heads might aid in better understanding the text-image relationship. Using a blending architecture can help reduce the amount of time needed to train generative networks with complicated data.

2. Related Work

Text to Image generative networks are used to generate images from descriptions provided by the users. The text description provided is used for training with generative networks for the generation of images associated with the description. A generator with this network produces images depending on the embeddings with the noise inputs. The discriminator determines if an image is real or false. Both the generator and the discriminator gets better with time.

The objective of the generator is to generate visuals that will deceive the discriminator to believing and classifying them as genuine. This was Ian Goodfellow's first approaches to generative networks [13]. Ian used deep convolutional neural network for creating generative networks. Deep CNN are used for building the generative networks [14].There have been numerous publications on generative networks, and GANs can now generate photorealistic images with extremely high resolutions. Recent research has focused on creating images from textual descriptions, but there are few unique solutions to this problem. As the initial answer to this challenge, Deep Convolutional GANs [2] were used to synthesise low-quality pictures from captions. This paper was unable to create images that were sufficiently realistic. Also, several synthetic photos did not quite fit the description. Author got inspired and created the Generative Adversarial WhatWhere Network (GAWWN) [15]. It displayed the bounding box of the object within the image and zoomed in on certain areas. It models the distribution of numerous elements, such as the beak,tail and feathers, in order to produce effective outcomes by concentrating on that region. This was useful for identifying crucial items for creation, but was unable to focus on key features such as stances or structure. It is possible to produce a picture more closely resembling the text description and to use this as a take off point to investigate text to image GANs. The problem of picture quality led to the creation of Stack GANs. This model used a layered method to increase image resolution at each stage and make 256-by-256-pixel photos that look real. The early model could create pictures with a resolution of 64 by 64 pixels. This method produced 64x64 pictures initially and got trained using generative networks in two phases to produce 128x128 resolution in the first stage, Stage-I. Procceding with 256x256 images in Stage-II. The objective of every stage was to increase the image quality and achieve a high quality impression. StackGANs [16] can produce lifelike pictures of birds and flowers. While lifelike pictures were produced, effective contextual extraction was absent. It lead to AttnGANs [7], which adopted a novel approach with a text-description-based attention mechanism. Attention capture allowed the network to get a deeper knowledge of context and generate visuals from text more effectively. AttnGANs utilised word embeddings and was able to extract key terms from bird descriptions in the CaltechUCSD Birds dataset. Attentional generative network was able to synthesise fine-grained features in distinct subregions of the image by paying close attention to the appropriate phrases within the text description. This work also developed a deep attentional multimodal similarity model (DAMSM) [7] to be the loss function, which is get assosiated with text description and generates picture features. To synthesise visual information, word-level conditional selection was implemented. For natural language processing, AttnGAN utilised bidirectional LSTM. Comparable to this study is Controlalable Text to Image Production, which successfully generate high quality photographs by managing features of the image generation pertaining to natural language descriptions [17]. This research employes channel-wise attention mechanism and a descriminator at the word-level along with the AttnGAN attention mechanism. The text-to-image synthesis resulted in a loss of perception. By upgrading the architecture of GANs by associating produced images with input descriptions, The approach using redescription has been implemented in MirrorGANs and with Cycle GANs utilising the BERT trasnformers model, and the paer suggested a significant performance boost on complicated datasets [18] [19]. Transformers model have recently enhanced natural language processing and are utilised by the majority of contemporary intelligent systems. While previous work has yielded excellent results, it is essential to comprehend how well these models will behave with the most recent transformer models. The RoBERTa model helped reduce the FID score from 20.77 to 23.98 in the main AttnGANs paper for the CUB Birds dataset as shown in Table 1. [6].

Table 1. Model Comparison for Text to Iage generation in current research.

Model	GAWWN [15]	StackGANs [16]	AttnGAN [7]	AttnGANs with RoBERTa [6]
FID Score	67.22	51.89	23.98	20.77

The images that are generated with our generative networks are individual images of each class based on their description. Table 2 tabulates all the research currently in Text to Image generation. The images need to be segmented so as to extract the mask of the object and then blend it with other image classes. [8] Mask R-CNN is a widely used image segmentation algorithm and helps extract the masked image. Feature maps are generated by running images through a CNN. Multiple Regions of Interest (RoI) are generated using a CNN and a lightweight binary classifier by the Region Proposal Network (RPN). It achieves this by placing nine anchor boxes on top of the image.

Classifiers provide object and no-object ratings. Network with the RoI allign has several bounding boxes instead of single definitive box which warps them onto a particular fixed dimension. Then, features warped are added to a fully-connected layer for classification using softmax. The regression model is used to adjust the boundary box prediction. Additionally, warped features are supplied to the mask classifier that consists of only two CNNs and generates a particulr binary mask assosiated for each RoI. The Mask classifier enables the particular network for production of the mask in each class without competition between classes. The original paper uses the COCO dataset as training data for segmenting images. [1] In this work, the Mask R-CNN is custom trained on bird and flower datasets. A mask generated from a bird is blended with the image of a flower. This is acquired using the Poisson blending technique. Finally, an image is created with both classes blended together at the scene based on the description. The description provides details like the position the object needs to be placed for blending.Principal advantages of this application include the usage of blended architecture so that several classes of images can be generated by generative networks and then blended together based on their respective descriptions. This will assist artists, game developers, and animation companies in creating scripted scenarios or provide them with a head start on the scene they wish to create.

Table 2. Related Works tabulated for Text to Image generation

SL.NO	TITLE	YEAR
1	Generative Adversarial Text to Image Synthesis [2]	2016
2	GAWWN (Generative Adversarial What-Where Network) [15]	2016
3	StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks [16]	2017
4	AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks [7]	2017
5	MirrorGAN: Learning Text-to-image Generation by Redescription [18]	2019
6	Controllable Text-to-Image Generation [17]	2019
7	Cycle Text-to-Image GAN [19]	2020
8	Text to Image GANs with RoBERTa and Fine-grained Attention Networks [6]	2021

3. Dataset

Experiments made use of the Caltech-UCSD Birds-200-2011 (CUB200-2011) image collection, which includes 200 different types of birds.There are 11,788 photos with annotations in all. The combined size of the photographs and annotations is around 1.1 GB. Birds dataset serves to be a standard for every research on text to image genertaive networks. Each of the photographs is accompanied by a boundary box of varying dimensions. This collection comprises photos of 200 distinct species of North American birds. The dataset (CUB-200) was compiled in 2010 and comprises around 6000 images for each of the 200 species of birds. This was complemented by additional label data, such as bounding boxes, rough segmentations, and extra characteristics. The dataset was updated in 2011 (CUB200–2011) to contain an additional 12,000 photographs, increasing the overall number of images to about 12,000. There are now 15 elements locations, binary characteristics of around 312, and particular bounding box for each picture among the properties that can be used. [3] A 102-category dataset comprising 102 flower categories is available. The selected flowers are those that are widely located in the United Kingdom. Each class is composed of 40 and 285 images. The images vary greatly in scale, position, and lighting. In addition, there are categories with substantial variance within the category and a number of categories with striking similarities. [4]

4. Algorithm

1. Input: Text description about bird, flower, background details, and positional Information
2. Extract details and information related to multiple classes:

   (a) Description of Bird.

(b) Description of Flower.
(c) Positional Information of Bird and Flower.

3. Repeat for each class:

(a) Word Embeddings and Sentence Embedding extracted using RoBerTa.
(b) Caption Feature input to Noise Vector
(c) Attention GANs are used in 3 Stages with Attention from word embeddings input at each stage to the Generator.
(d) Generate 256 X 256 Images of that class.
(e) Image Passed to CNN and features decoded to generate Image Embeddings.
(f) DAMSM is calculated and used as a loss function.
(g) Stop after 600 epochs and final 256 X 256 Image Generated.

4. Input each image with Positional and background information to custom Mask R-CNN model for image segmentation.
5. Segmented images generated are blended using the positional information and background features with Poisson Blending.

5. Method

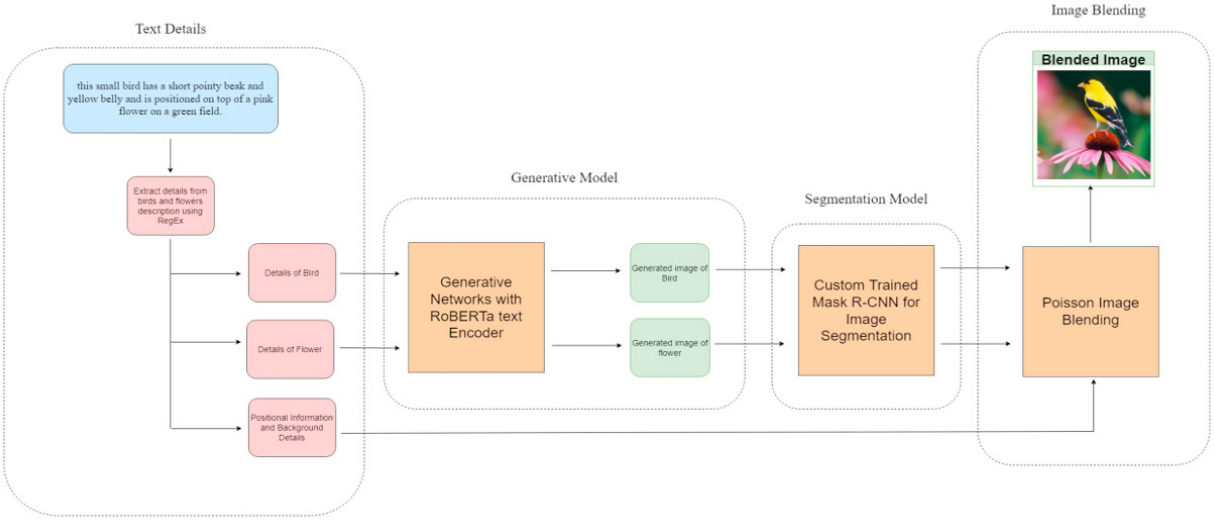

Fig. 1. Application workflow from text input from user to image generation by the system.

Fig. 1, illustrates the application's architectural flow. The user provides the description, which comprises information about various classes. Each class description is uniquely submitted to and trained by the generative network. In the blending section, both the positional information and background information for each class are supplied. Training in isolation can reduce total training time in comparison to training with mixed classes in generative networks, such as the COCO dataset. In this study, birds and flowers serve as the demonstration use case. Thus, we separately train the birds dataset and the Flowers dataset. In generative networks, text is sent to a neural language processing network that employs the RoBERTa neural language model to interpret the text description. Text and picture datasets are used to train generative networks. The generative network employs DAMSM as the loss function and is trained for 600

iterations. The generative networks generate a picture from a textual description. An image of a bird and an image of a flower are created separately. For image segmentation, the Mask R-CNN is utilised, and pictures are segregated for each class. In this article, tailored instruction for each class is provided. The retrieved mask picture is now mixed based on the location information and backdrop features. Poisson blending [12] is used to mix images. Thus, a new picture matching the basic description is being created.

5.1. Attention GANs with RoBERTa neural language mode

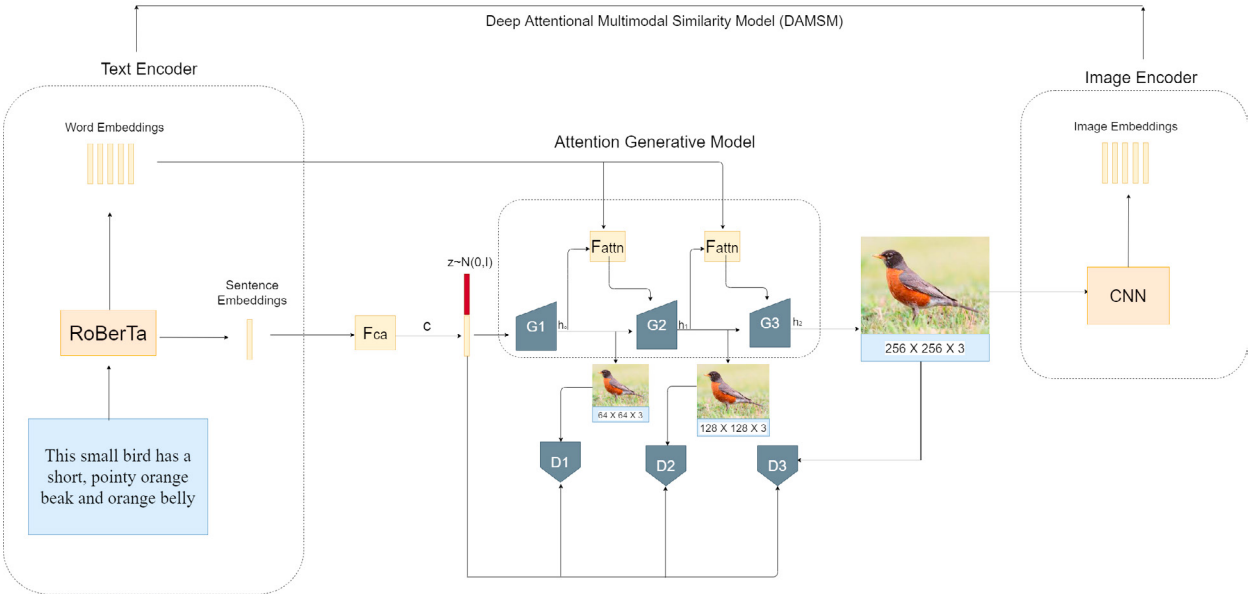

Fig. 2. Architecture of the attention generative network with RoBERTa language neural model.

Fig. 2 depicts the architecture for attention generative networks using the RoBERTa transformers language model [6]. AttnGAN utilizes attention element which takes the described caption in the objects dataset and passed on to RoBERTa transformers model in so on to build words and phrases vectors. Captions used by the text encoder, that is a sentence of T-word , as an input. the characteristics of sentence contributes to the global vectors, which gets transmitted to noise vector. Sentence feature makes up final hidden dimension D state. The word features are separately extracted, which creates a state which is concealed from every T-word sentence timesteps.

$$\bar{e} \in \mathbb{R}^{D} \tag{1}$$

$$e \in \mathbb{R}^{DXT} \tag{2}$$

The conditioning enhancement draws latent variables at arbitrary in the Gaussian distribution.As a result, $\bar{e}$, that is an input to the caption feature, μ as well as σ are segregated into and utilising a densely integrated linear layer. That is the standard deviation and mean of the embedded sentence data. The resultant mean as well as variance are being taken to parameterize normal distribution where a sample phrase embedding is created for such generative network,which gets coupled along a noise vector to generate photos with increased diversity for one description. The c vector gets merged along the Z noise component, and this is utilised in subsequent stages to produce a variety of bird properties within the network. Likewise, word features are collected individually. This generates a state to the T-word phrase that is concealed from all timesteps.

$$\begin{aligned} &\bar{e} \longrightarrow \mu, \sigma \\ &c = \mu + \sigma * \varepsilon, \varepsilon \sim N(0, I) \end{aligned} \tag{3}$$

The primary responsibility of the first generative network is upsampling. With a scale factor of 2, the closest neighbour interpolation is employed to upsample the data. The result consists of 64x64 resoolution image. This do not employ the word-level characteristics derived by the RoBERTa transformer model. the model employs the sentence-level features that are extracted from vector space of the noise.

$$h \in \mathbb{R}^{\hat{D} \times N}$$
$$h_0 = F_0\,(z, F^{ca}(\bar{e})) \tag{4}$$

. The initial attention network incorporates e, that is the word assosiated features with context from preceding stage h_{i-1}. Word features are integrated into a shared space which is e'. This is achived by incorporating a perceptron layer and is represented using $e' = Ue$, given that $U \in \mathbb{R}^{\hat{D} \times D}$. Each column assosiated with h is a vector of image features for a subregion. It creates a score for a particuar subregion j and word I by combining them with the context.

$$s'_{j,i} = h_j^T e'_i \tag{5}$$

$$c_j = \sum_{i=0}^{T-1} \beta_{j,i} e'_i, where \beta_{j,i} = \frac{\exp\left(s'_{j,i}\right)}{\sum_{k=0}^{T-1} \exp\left(s'_{j,k}\right)} \tag{6}$$

As a result, the word-context vector for each subregion has been built using a subregion-specific word fusion. All region goes through the same process. In this case, the attention network's output is produced.

$$F^{attn}(e, h) = (c_0, c_1, \ldots, c_{N-1}) \in \mathbb{R}^{\hat{D} \times N} \tag{7}$$

The second generator is likewise used to upsample the image, producing a 128x128 image. In this step, in addition to the prior output to input which is taken from the initial generator that has the context vectors, the embeddings of words through the attention element carrying their word context vectors are also included. The residual blocks in this case deepen the network and train it without degradation. Comparable to the second generator, a third generator is assosiated with the upscaling of image to 256x256 resolution and it accepts similar input. Ultimately, a 256x256 resolution image is sent to the image encoder. Local features of image could be extracted in the image encoder and then transformed to a common space that matches with the text encoded features. Together, they form the Deep Attentional Multimodal Similarity Model (DAMSM), which gets trained through attention loss. Stability of system improves through pretained DAMSM loss. Three discriminators are connected to their respective generators. As input for each discriminator, the sentence level features are used by avoiding noise vector. The network uses two forms: an unconditional form, which indicates if the generated image is true to the ground truth or false, along with conditional form, which indicates whether the genrated image and description pair are the same. In the unconditional pair, if both pairs match, a result close to 1 is produced. The RoBERTa text encoder employs transformers with an attention mechanism that discovers the contextual relationship between the words in a sentence. Attention GANs utilise the fundamental RNN, a bidirectional LSTM. In order to get the semantic vectors, LSTM is applied to the textual description. [7]There are two hidden states for every word for bidirectional LSTMs. For example, Facebook's RoBERTa: A Robustly Optimized BERT Pretraining Approach, enhances the BERT functionality. With dynamic masking, a masked token varies all across the course of the training sessions, making it easier to remember. Roberta-base' prediction model is employed in this instance. The embedding dimension used during training stage was 768. As part of the Attention GAN architecture, the pre-trained RoBERTa model is used to link the word and sentence embeddings. T he extracted features are mapped to a shared space by using the Image encoder in combination with DAMSM. Using a CNN, the image's subregions can be learned by the intermediate layers, whereas the image's global features is learned in the final layer. ImageNet's Inception-v3 model is being used for image encoding. The local feature dimmension is set around 768, and the resizing of image is done with a dimmension of 299x299 pixels to generate 289 subregions. By integrating perceptron layers, these attributes are ultimately translated to a comparable space for that of a text encoder. With each G_i, generation a D_i discriminator, loss equal sum of conditional and unconditional at every stage.The discriminator receives the generated pictures sampled from the generator of the provided distribution through the un-

conditional loss. The loss is reduced so the discriminator believes the incoming image is real. For the conditional loss, the discriminator was provided with $\bar{e}$ and the resulting image.

$$\mathcal{L}_{G_i} = \underbrace{-\frac{1}{2}\mathbb{E}_{\hat{x}_i \sim p_{G_i}}\left[\log\left(D_i\left(\hat{x}_i\right)\right)\right]}_{unconditionalloss} \underbrace{-\frac{1}{2}\mathbb{E}_{\hat{x}_i \sim p_{G_i}}\left[\log\left(D_i\left(\hat{x}_i, \bar{e}\right)\right)\right]}_{conditionalloss} \tag{8}$$

Cross-entropy loss is used to discriminate between original and created distributions with in discriminator. There's a good chance that a discriminator's loss will be minimised if it can get the original distribution and the images it generates near to 1 and generated image's output near to 0.

$$\mathcal{L}_{D_i} = \underbrace{-\frac{1}{2}\mathbb{E}_{x_i \sim p_{data_i}}\left[\log D_i\left(x_i\right)\right] - \frac{1}{2}\mathbb{E}_{\hat{x}_i \sim p_{G_i}}\left[\log\left(1 - D_i\left(\hat{x}_i\right)\right)\right]}_{unconditionalloss} + \underbrace{-\frac{1}{2}\mathbb{E}_{x_i \sim p_{data_i}}\left[\log D_i\left(x_i, \bar{e}\right)\right] - \frac{1}{2}\mathbb{E}_{\hat{x}_i \sim pG_i}\left[\log\left(1 - D_i\left(\hat{x}_i, \bar{e}\right)\right)\right]}_{conditionalloss} \tag{9}$$

5.2. Mask R-CNN

For image segmentation and instance segmentation, Mask RCNN is a state of the art Convolutional Neural Network (CNN) model. A Region Based Convolutional Neural Network called faster CNN is used in this network. When using Mask R-CNN, a third branch is added that outputs the object mask, while Faster R-CNN just offers the class label as well as the bounding-box offsets. The extra mask result is distinct from the outputs of the classes and the boxes. This necessitates a considerably more exact extraction of the object's spatial configuration. In addition to the branches for identifying the bounding box the R-CNN extends Faster R-CNN by predicting an object mask which is the Region of Interest. Unlike Mask R-CNN, Fast R-CNN will not have the pixel-to-pixel synchronization that is essential for Mask R-CNN. This approach is similar to Mask R-CNN, which employs a similar two-step procedure with a similar first stage which is a Region Proposal Network. Mask R-CNN generates a binary mask for every ROI in conjunction to class and box offset predictions in the second phase. [8].

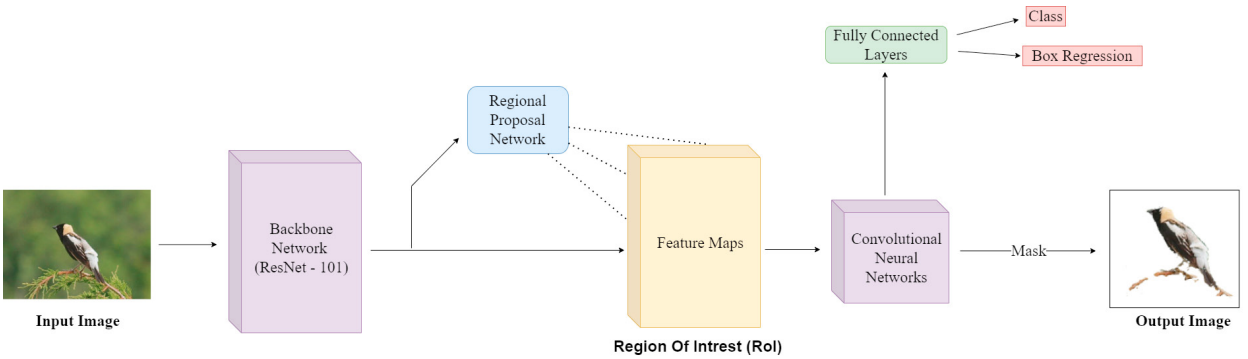

Fig. 3. Custom trained Mask R-CNN with bird image as input as mask generation as output.

The input image of a bird is passed to a pretrained convolutional network as shown in Fig. 3. In this paper, a 101-layered Resnet model has been used as the backbone network. The CNN extracts features from the image input, and these features are input for the Regional Proposal Network. The RPN has two convolutional layers in which one acts as a classifier and the other as a regressor. A Region Proposal Network (RPN) generates several Regions of Interest (RoI) by employing a CNN and a lightweight binary classifier. This is achieved by placing nine anchor boxes over the image. Mask R-CNN uses anchor boxes to acknowledge numerous, overlapping, and varying-sized objects. This improves speed and accuracy. Anchor boxes are pre-sized bounding boxes. These boxes record the size and aspect ratio of specified object types. The classifier gives object/no-object scores. Mask R-CNN makes hundreds of predictions in order to anticipate numerous objects or instances of things in an image. Final object identification is done by eliminating background anchor boxes then filtering residual anchor boxes using confidence score. The anchor boxes with IoU ¿ 0.5 is identified. Non-Max suppression selects the most confident anchor boxes. The binary

classifier informs whether this pretrained network's feature map includes the bird . The regressor draws a bounding box when an item is identified. RPN creates areas of interest that are provided to ROI Pooling together with pretrained network feature maps. The RoI Align network produces many bounding boxes and compresses them into a fixed dimension. Various sizes and aspect ratios are retrieved. This produces differing feature dimensions and should be processed to rectify them which is proccessed by the RoI pool. The fully linked layer will receive ROI pooling's output. The classifier provides the image's class using softmax. A bounding box is plotted over the detected bird. The mask classifier, which consists of two CNNs to generate a binary mask for each RoI, is also fed warped features Segmentation [9] [10] [11] of image is obtained along with the mask for Mask R-CNN. The Mask Classifier makes it possible for such network to build masks for every class even without conflict between classes. Lastly, the mask of the object, which is flower here is generated that is utilized for image blending.

5.3. Poisson Blending

Poisson blending is an image blending technique in which a user may merge two images together without generating any aesthetically unpleasant seams [12]. The hue of the added picture is also altered, so that the inserted item feels as if it is part of the target image's surroundings as well. So, if you copy and paste a bright item into a dark picture, the object's colour will be darkened. In this paper, we use flower and bird pictures generated from generative networks. The image of a flower is copied and pasted into an image of a bird using poisson blending. The concept behind poisson blending is straightforward: we encode the target picture as gradients, but when we want to copy and paste the source image, we just copy and paste the source image's gradients. Thus, in the region affected by copy-and-paste, the original gradients are replaced with the source gradients, and a picture is reconstructed using these gradients. Despite the fact that poisson blending alters the colour of the source image, it maintains its characteristics. With poisson blending, the gradients of one image are pasted into the other, and as a result, the solver is not always able to recover an image whose gradients precisely match the prescribed gradients. But the solver attempts to find a picture whose gradients match as closely as possible, and in reality, poisson blending provides good results.

6. Platform and Specification

The experiments were conducted utilising a cloud-based Google Collab Pro membership. Tesla V100 GPU with 16GB VRAM and 24GB CPU RAM was utilized. On NVIDIA RTX 3060 GPUs with 6GB VRAM and 16GB CPU RAM, a few local tests were also conducted. The entire code was performed using Pytorch library with python 3.7 as the programming language.

7. Experiments and Results

For this paper RoBERTa pretained language model from huggingface library was used [6] with the attention generative networks that ran DAMSm loss. It took up to 600 epochs of training for each bird as well as flower data set separately in generative network. To guarantee training stability, overall batch size was kept at 48 and learning rate at 0.00005 was kept at encoder. Gradient clipping is set up at 0.25. The latent structure for text embedding included 768 dimensions, and that for training, 10 captions for every picture were gathered. To obtain the attention map established when pretraining DAMSM using the pretrained RoBERTa text encoder, along the base size is set to 299 characters. A RoBerTa tokenizer as well as a pre-trained 'roberta-base' model were used to train the transformer. The ImageNet's Inception-v3 model served as the foundation for CNN's first training. In Fig. 4, an attention map is shown that was derived from the produced bird picture. Same goes for Fig. 5's flower generated map. In each frame, text is connected with a section of the class, and that portion of the class is then captured. The hue allocated to a body component is determined by the linguistic model. Attention outlines how much each word pertains to synthesising a specific part of the objects picture. Once the pre-training is completed, the DAMSM approach yields text encoders and image encoders. AttnGANs architecture training utilizes this. The RoBERTa encoder was used to train the AttnGAN network for 600 epochs. (GF_DIM) and (DF DIM) were limited to 32 convolutional filters in the first layer of the generator and discriminator, respectively, due to resource constraints.

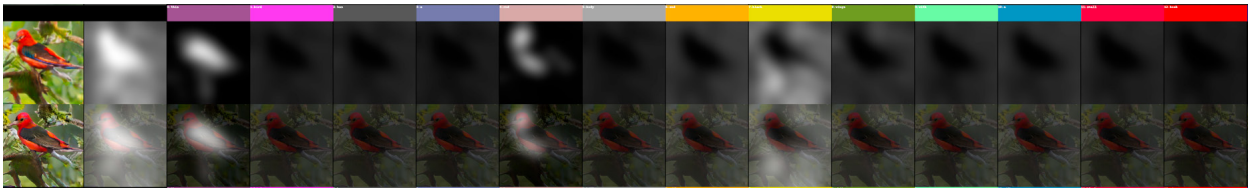

Fig. 4. Attention maps generated based on text description for the bird generated image.

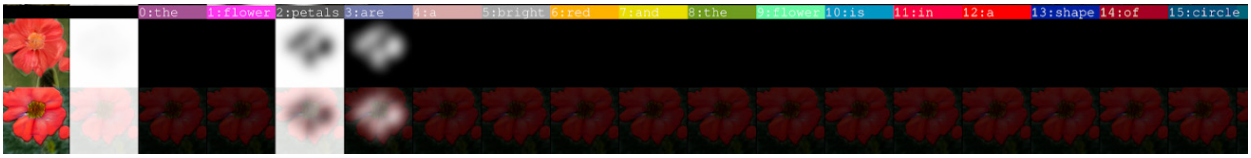

Fig. 5. Attention maps generated based on text description for the flower generated image.

Eight batches per epoch was used for training. The generator and discriminator's rate of learning was fixed at 0.0002 The RoBERTa text encoder has a resolution of 768 by ten captions for each image. The noise vector models dimension count was maintained at 100 throughout the training procedure. Within 600 epochs, images close to real object were created and the model is saved for later application. With a more powerful GPU, more generative networks can be utilised to turn the image to a higher quality. The model generated Fig. 6 after roughly 600 epochs. The 'roberta-base' model was utilised to extract the text description's context. The primary objective of the natural language model is to identify attention heads and establish bidirectional word associations. Fig. 6 shows the result of image generation for the text description of a bird: "This bird has a pointy tail and beak with brown body and a white belly", and the image generated by providing the flower description: "This flower is pink and yellow in colour with petals that are oval shaped". Both were trained for 600 epochs. The image dataset of birds is better quality and hence the GANs could be trained to generate good photorealistic images. This is the reason the flower images generated aren't as clear as birds as the dataset was of lower quality. An application was developed on bird generative networks for the quick generation of bird images. This is a web application developed using the Streamlit library and uses the 600 epoch trained model, which is pretrained with DAMSM loss and uses the RoBERTa text encoder. The screenshot of the application is shown in fig. 7. The application also catches and visualises the attention mechanism happening behind the scenes. This application depicts the AttnGANs with the RoBERTa module. The output produced from here is sent to Mask R-CNN for generating the image mask. The Mask R-CNN is a pretrained model on the COCO dataset that is used for Mask generation of images [8]. For this experiment, the Mask R-CNN is customly trained on both birds and flower dataset. From each dataset, 600 images were randomly selected for this experiment. The backbone network used in our model was Resnet101 architecture which is a pretrained model with strides of [4, 8 , 16, 32, 64]. For training purpose a batch size of 2 image were used. This means at each iteration two images were loaded in the GPU. The boundry box standard deviation was set to [0.1, 0.1 ,0.2, 0.2] with a maximum instance of 100 was set. The DETECTION_MIN_CONFIENCE is provided as 0.9 which skip detections with less than 90 percentage confidence. The IMAGE_MAX_DIM is set to 1024 and IMAGE_MIN_SIZE to 800. All the images provided are reshaped to the IMAGE_SHAPE of [1024, 1024, 3]. A learning rate of 0.0001 is set in the network. The Mask generated is in the shape of [28, 28]. The setting of Max ground truth instances, MAX_GT_INSTANCES is set to 100. A pretrained COCO dataset can also be used directly for this purpose. Mask-RCNN vastly simplifies these tasks: the existing bounding-box prediction that is the localization task. The head predicts the class, similar to faster-RCNN, while the mask branch generates a mask for each class. The utilised loss is per-binary loss + pixel sigmoid . The loss generated in the network for birds and for flowers is depicted in fig. In case of the bird, the network seems to generalize well after 40 epochs with a loss of 0.1. In case of flowers it only seems to generalize after 70 epoch with a loss around 0.03 as shown in Fig. 8. The loss here is sum of all individual losses. In case of bird dataset, the images were more clear with better resolution and background with object identification was more easier hence. In case of flowers, images arent of good resolution as birds and were bit unclear. This would have caused the model to train more to identify the object and differentiate it with the background.

Fig. 6. (a) The generative network produced Image of a bird for the caption 'This bird has a pointy tail and beak with brown body and a white belly'; (b) Image generated by the RoBERTa GANs for the caption 'This flower is pink and yellow in colour with petals that are oval shaped'.

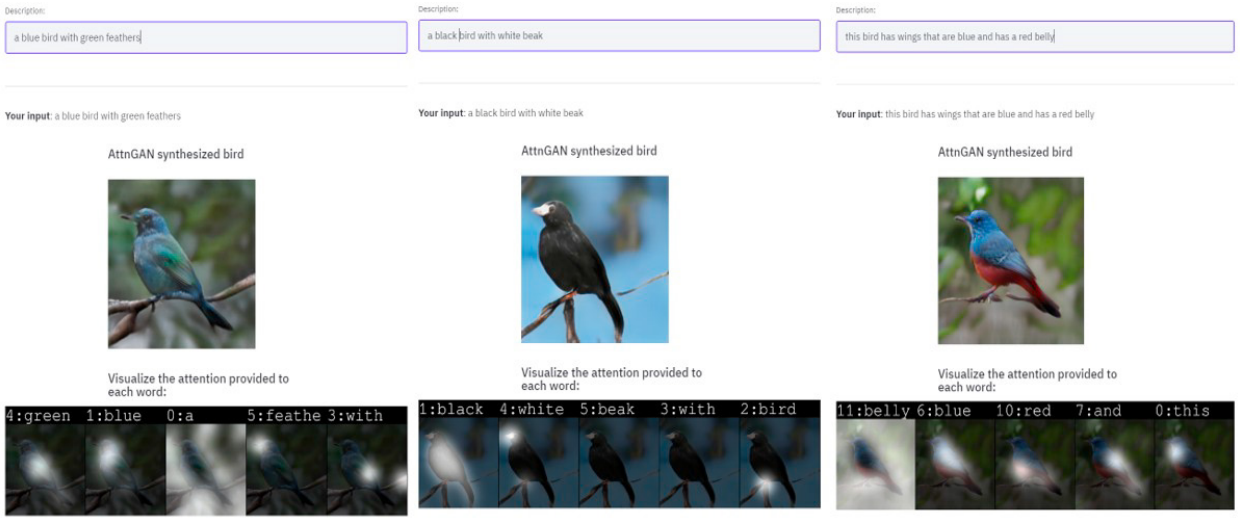

Fig. 7. Streamlit application screenshot for generating birds image quickly using AttnGANs with RoBERTa trained model.

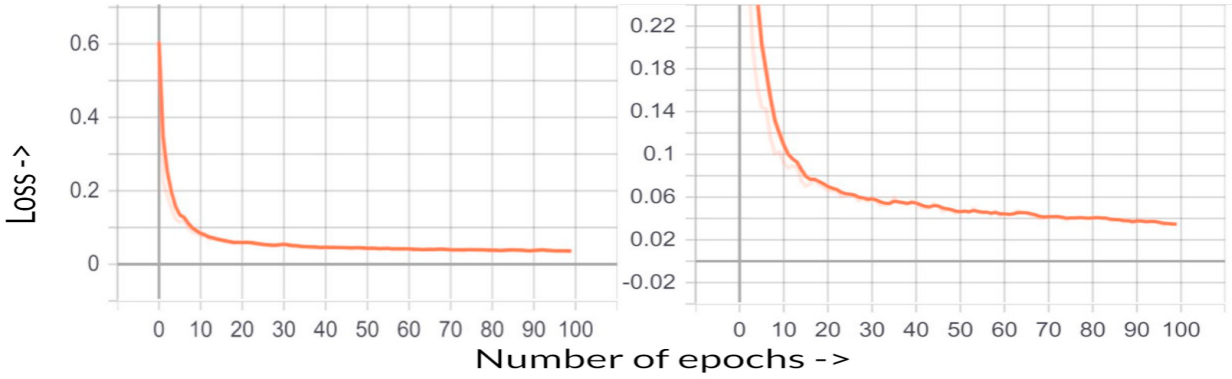

Fig. 8. a) Loss generated by Mask R-CNN on training Caltech CUD Birds dataset (600 random sample instances); b) Loss generated by Mask R-CNN on training Oxford 102 Flower dataset (600 random sample instances)

The mask generated from the flower model is then used to blend with the image of the bird. A specific mask from a bird and flower could be individually generated and merged with the details like background details and position provided externally. In this experiment, the mask is generated from flowers and merged directly with the bird image generated by GANs. Positional details are provided by users externally, and the mask generated is placed according to the description of the user. Fig. 9 shows some results of the application developed.

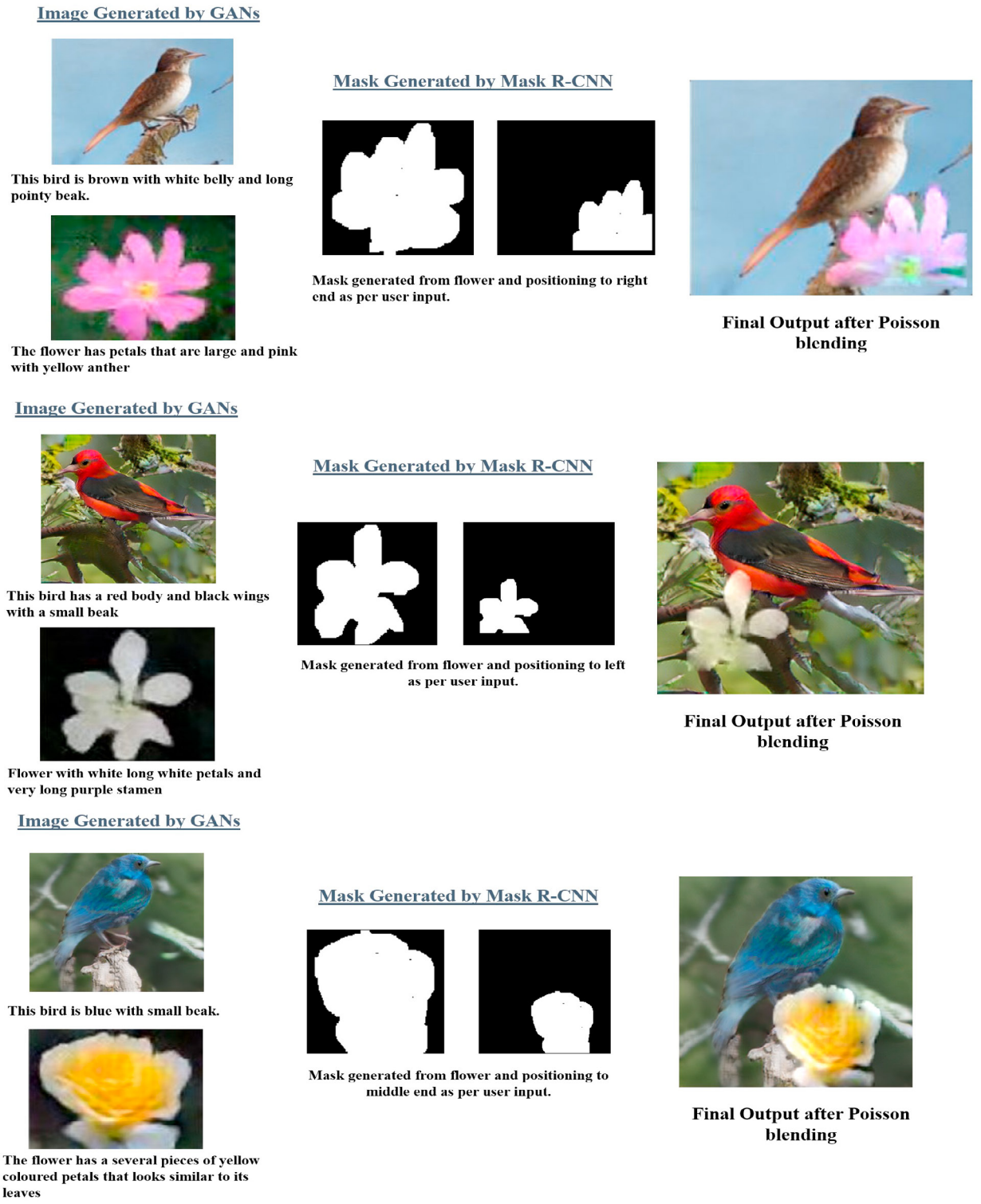

Fig. 9. Blended multiclass image of bird and flower generated by generative networks along with the mask of flower dataset generated using Mask R-CNN.

8. Conclusion

This paper provides an application that can help artists, game developers, and movie creators get a head start in generating an image for a scene based on their description in mind. The user provides the description of different classes along with the positional and background details of the system. The description of each class is input to the generative networks. The generative network receives captions at the level of sentences and words embeddings and leverages the use of latent space in the noise vector so on to generate images of birds and flowers that correspond to the text description provided by the user. Using DAMSM, the loss of fine-grained image to text matches was identified. This loss is then applied to training of the generator. Inception V3, a pre-trained model was utilisedand by the image encoder and the text encoder utilised RoBERTa transformers pre-trained model. The resulting image generated is passed to the Mask R-CNN network, which processes the image to generate the mask. In our experiment, we used the Caltech Birds and Oxford Flowers datasets. For masking, we used the flower generated by GANs, and its respective image was segmented and a mask was generated. This was blended with the bird image generated by the generative networks. Finally, an image is generated with the blending of multiple classes together. Training a generative network individually with all classes and trying to recreate scenes could be very complex and time-consuming. With this architecture, a head start is provided for the creation of scenes with the classes trained in isolation by the generative network.

References

[1] Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft coco: Common objects in context. InEuropean conference on computer vision 2014 Sep 6 (pp. 740-755). Springer, Cham.

[2] Reed S, Akata Z, Yan X, Logeswaran L, Schiele B, Lee H. Generative adversarial text to image synthesis. InInternational conference on machine learning 2016 Jun 11 (pp. 1060-1069). PMLR.

[3] Welinder P, Branson S, Mita T, Wah C, Schroff F, Belongie S, Perona P. Caltech-UCSD birds 200.

[4] Nilsback ME, Zisserman A. Automated flower classification over a large number of classes. In2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing 2008 Dec 16 (pp. 722-729). IEEE.

[5] Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692. 2019 Jul 26.

[6] Siddharth M, Aarthi R. Text to Image GANs with RoBERTa and Fine-grained Attention Networks. International Journal of Advanced Computer Science and Applications. 2021;12(12).

[7] Xu T, Zhang P, Huang Q, Zhang H, Gan Z, Huang X, He X. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. InProceedings of the IEEE conference on computer vision and pattern recognition 2018 (pp. 1316-1324).

[8] He K, Gkioxari G, Dollár P, Girshick R. Mask r-cnn. InProceedings of the IEEE international conference on computer vision 2017 (pp. 2961-2969).

[9] Sikha OK, Kumar SS, Soman KP. Salient region detection and object segmentation in color images using dynamic mode decomposition. Journal of Computational Science. 2018 Mar 1;25:351-66.

[10] Subbiah U, Kumar DK, Thangavel SK, Parameswaran L. An extensive study and comparison of the various approaches to object detection using deep learning. In2020 International Conference on Smart Electronics and Communication (ICOSEC) 2020 Sep 10 (pp. 183-194). IEEE.

[11] Aloysius N, Geetha M. A review on deep convolutional neural networks. In2017 international conference on communication and signal processing (ICCSP) 2017 Apr 6 (pp. 0588-0592). IEEE.

[12] Pérez P, Gangnet M, Blake A. Poisson image editing. InACM SIGGRAPH 2003 Papers 2003 Jul 1 (pp. 313-318).

[13] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. Advances in neural information processing systems. 2014;27.

[14] Aarthi R, Harini S. A survey of deep convolutional neural network applications in image processing. Int. J. Pure Appl. Math. 2018;118:185-90.

[15] Reed SE, Akata Z, Mohan S, Tenka S, Schiele B, Lee H. Learning what and where to draw. Advances in neural information processing systems. 2016;29.

[16] Zhang H, Xu T, Li H, Zhang S, Wang X, Huang X, Metaxas DN. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. InProceedings of the IEEE international conference on computer vision 2017 (pp. 5907-5915).

[17] Li B, Qi X, Lukasiewicz T, Torr P. Controllable text-to-image generation. Advances in Neural Information Processing Systems. 2019;32.

[18] Qiao T, Zhang J, Xu D, Tao D. Mirrorgan: Learning text-to-image generation by redescription. InProceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2019 (pp. 1505-1514).

[19] Tsue T, Sen S, Li J. Cycle text-to-image GAN with BERT. arXiv preprint arXiv:2003.12137. 2020 Mar 26.