TEXT TO IMAGE GENERATION USING GAN

Mrs. L. Indira[1], M. Sunil[2], M. Vamshidhar[3], S. Ravi Teja[4], R. V. Praneeth[5]

[1] Assistant Professor, Department of Computer Science and Engineering
[2,3,4,5] Undergraduate Student, Department of Computer Science and Engineering, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, India

---***---

Abstract - *Producing good images from descriptions is a challenge in computer vision with practical applications. To address this issue, we propose Stacked Generative Interconnected Networks (StackGAN) to combine the 256×256 real images described in the annotation. The process splits the problem into two phases: Phase-I generates the low-level image by drawing pictures and colours from the text, while Phase-II edits the Phase-I results to create the high image with photorealistic details. . price picture. Conditional magnification is used to improve image contrast and stable GAN training. Extensive testing demonstrates that our development method is better than the state-of-the-art method and proves its effectiveness in creating realistic images based on descriptions. In summary, StackGAN provides a multi-level approach with additional optimization to improve composite images and shows great results in creating high-quality images from text*.

***Key Words*: Stack GAN, Resolution, Conditioning Augmentation, Image generation**

1.INTRODUCTION

Creating realistic images from descriptions is an important and difficult task, applicable in many fields such as photo editing and computer aided design. Recently, prolific competing networks (GANs) are showing promise in creating real images. Conditional GANs specially designed to generate images from descriptive text have been shown to be able to generate text-related images.

However, training GANs to create realistic images from annotations remains challenging. Increasing the number of layers used to solve image problems in current GAN models often makes training unstable and ineffective. The problem arises because the classification of the natural image and the implicit classification model do not overlap well in high pixel space. This problem becomes more serious as the image resolution increases. Previous tutorials were limited to creating sensible 64×64 images from descriptions without real objects and details like a bird's beak and eyes. Also, annotations are required to create higher resolution images such as 128×128.

To solve these problems, we present Stacked Generative Adversarial Networks (StackGAN), which decomposes the problem of text-to photorealism image synthesis into two manageable problems. We first developed the decoder using a level-I GAN. Next, we set the Level-II GAN on top of the Level-I GAN to create a really good image (for example, 256 × 256) according to the Level-I results and description. Leveraging the Tier-I results and text, the Tier II GAN learns to capture information the Tier-I GAN would miss and add more detail to the product.

This approach improves the rendering of high-resolution images by increasing the ability to model the distribution of the image distribution.

We also propose developing a new method to handle the differences in various text events caused by the limitation of the number of training text-image pairs. This technique facilitates smoothing of the central cooling manifold by allowing small random perturbations. It improves the contrast of synthetic images and improves the training of GANs. Our contributions can be summarized as follows:

(1) We propose a common crosslinking algorithm that improves the actual (256 × 256 resolution) efficiency of image binding without decomposing the problem into control problems.

(2) We introduce the amplification technique leading to different design and stability for training GANs.

(3) Through extensive and thorough testing, we demonstrate the effectiveness of the overall design and the impact of individual products. Such useful information could guide the GAN design pattern in the future.

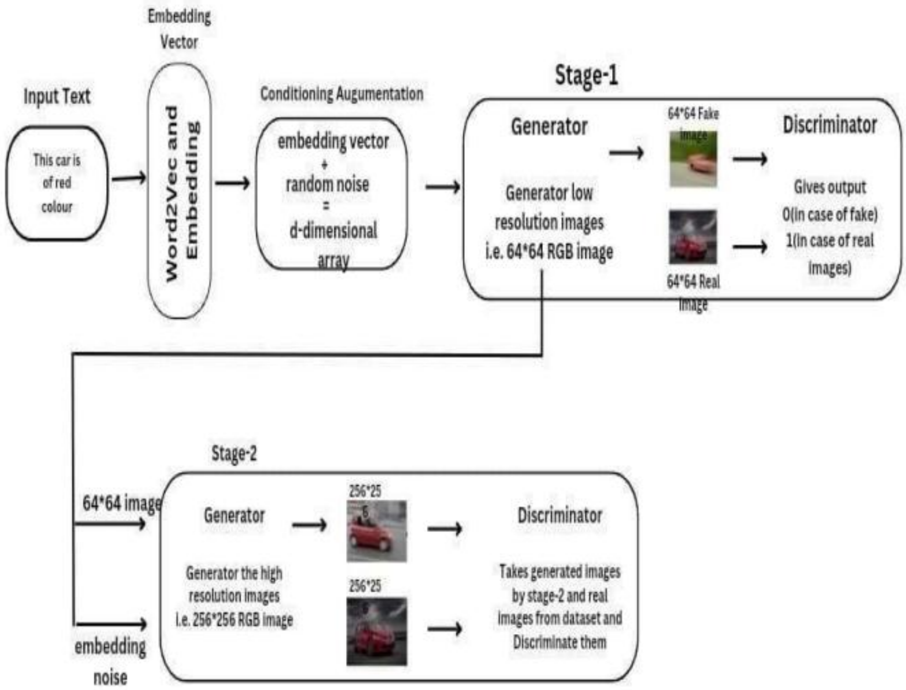

Fig -1: System Architecture

2. RELATED WORK

To solve this problem, researchers have proposed many ways to improve the training process and improve image quality. Some methods focus on power-based GANs, while others involve variables such as attributes or class records. There are also methods that use cool image-to-image functions such as photo editing, relocation, and super resolution. However, these super-resolution techniques have limitations in adding important details or correcting imperfections in low-resolution images.

Recently, attempts have been made to create images from negative descriptions.

These include methods such as AlignDRAW, event PixelCNN, and approximate Langevin sampling. While these methods have been shown to be effective, they often involve repetitive processes or have limited functionality.

Conditional GANs were also used to generate images from the annotations. For example, Reed et al. Reliable 64×64 bird and flower images were successfully processed using GAN formalism.

In the next work, they extended the 128×128 rendering method using annotation tools.

There are methods that use a series of GANs to create images, rather than using a single GAN. Wang et al. S2-GAN is proposed, which separates the internal scene production into model and style production. In contrast, the second level of StackGAN aims to improve the content of the product and fix the defects according to the description.

In summary, work on text-to-image synthesis using StackGAN leads to advances in modeling, including VAEs, autoregressive models, and GANs. Various methods have been proposed to stabilize GAN training, including adaptive conversion and rendering from annotations. While previous methods have shown good results, StackGAN stands out by tackling the challenge of creating high-resolution images with realistic details. Its bi-level architecture and conditional magnification techniques help synthesize diverse and visually appealing images.

3. METHODOLODY

Our proposed StackGAN approach addresses the challenge of creating high-quality images from annotations. To achieve this goal, we propose a two-step approach that breaks down the problem into more manageable problems and allows for the creation of realistic images from the descriptions. In the first phase of StackGAN, we use a generative competitor network

(GAN) to generate a low-resolution image from a given definition. This first phase, called Phase-I, focuses on capturing the shape and color of the objects described in the text. By creating low-resolution images, we lay the foundation for further development at higher levels. Stage-I GAN generates preliminary results from low resolution images as input for the next stage.

Our proposed StackGAN approach addresses the challenge of creating high-quality images from annotations. To achieve this goal, we propose a two-step approach that breaks down the problem into more manageable problems and allows for the creation of realistic images from the descriptions. In the first phase of StackGAN, we use a generative competitor network (GAN) to generate a low-resolution image from a given definition. This first phase, called Phase-I, focuses on capturing the shape and color of the objects described in the text. By creating low-resolution images, we lay the foundation for further development at higher levels. Stage-I GAN generates preliminary results from low resolution images as input for the next stage.

In training, we do many experiments using test data and compare them with the latest methods. These experiments are used to validate the effectiveness of StackGAN in generating realistic images in descriptive text. We considered both quality and quantity in our analysis.

For quality evaluation, we analyze the visual quality of the generated image and compare it with the actual image on the ground. We evaluate factors such as image clarity, accuracy of the description provided, and availability of good content and spare parts.

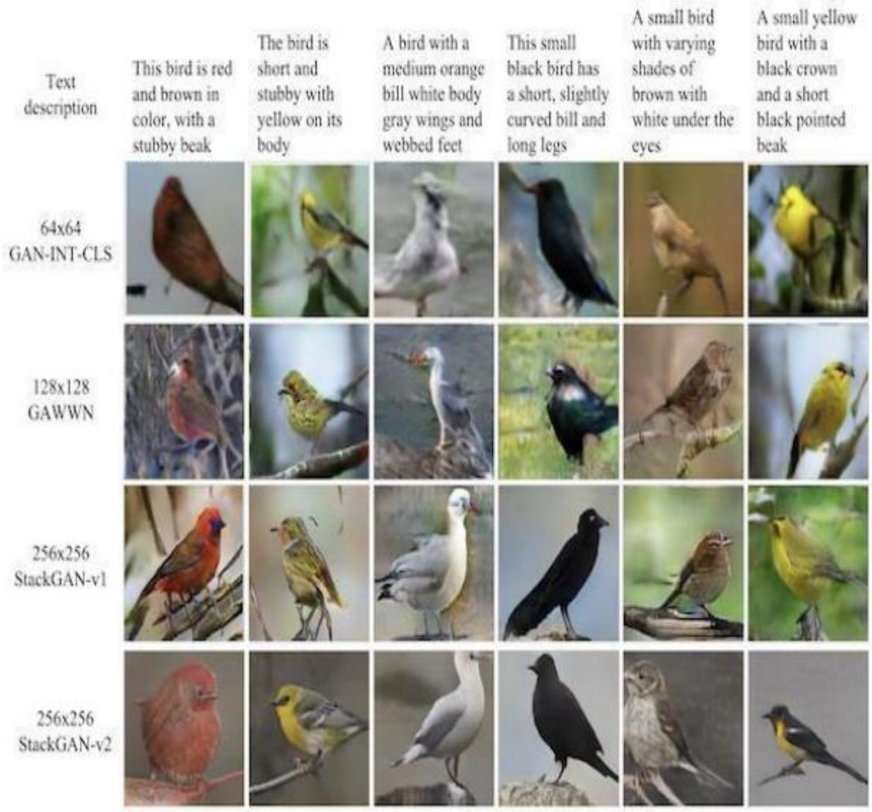

Fig -2: Comparison of the StackGAN images with other models

4. RESULTS

This project's major objective was to generate the image upon given text. The images which are generated during the training phase are as follows:

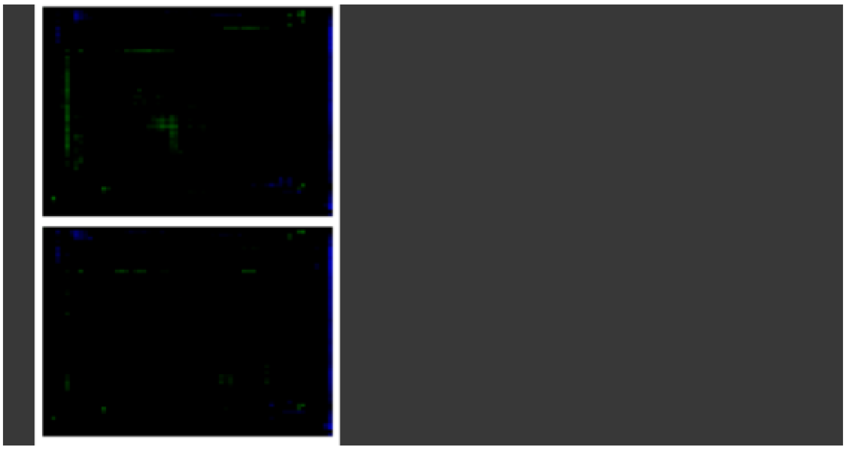

Fig -3: Training Phase Image

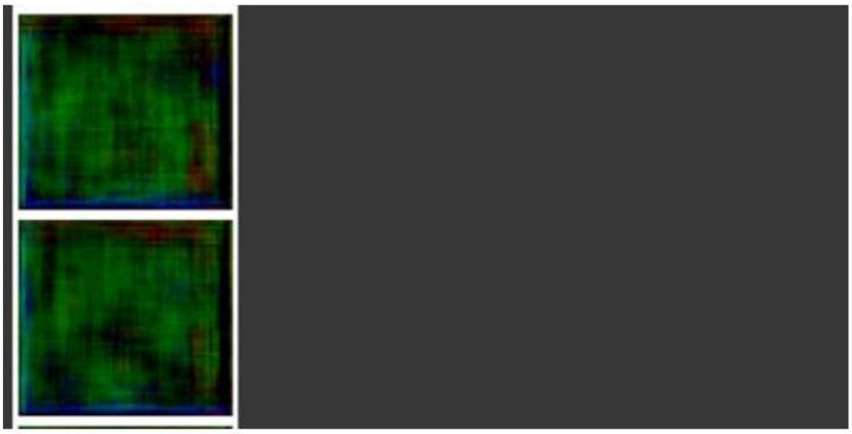

Fig -4: Taining Phase Image

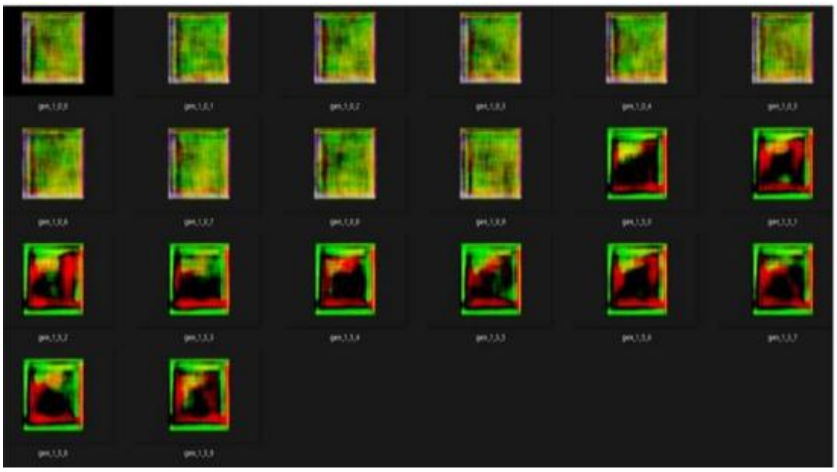

Fig -5: Training Phase Image

The output images when the text input is provided are shown below:

INPUT TO MODEL: Boy Riding a horse

OUTPUT OF THE MODEL:

Fig -6: Output Image

INPUT TO MODEL: A dog with wings flying in air.

OUTPUT OF THE MODEL:

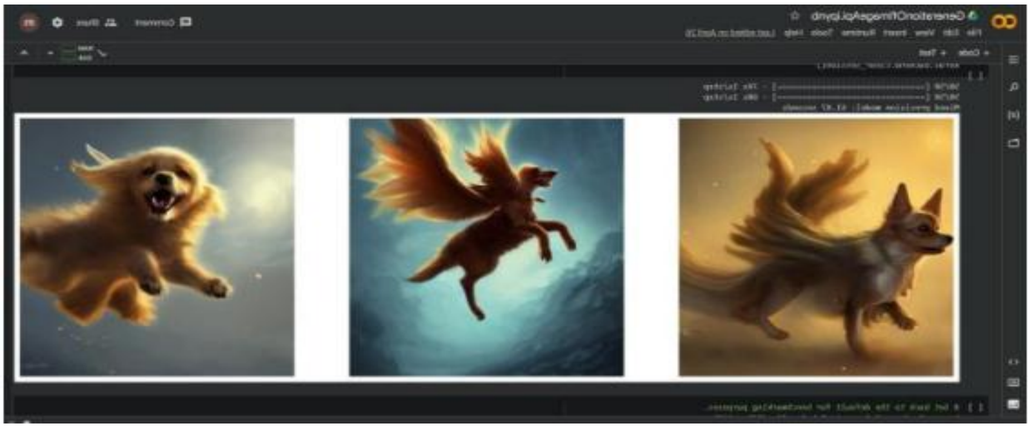

Fig -7: Output Image

Another final output image which is converted using GAN technology:

Fig -8: Output Image

3. CONCLUSIONS AND FUTURE SCOPE

In this work, we introduce a new method, StackGAN, to combine real time images from annotations using linear compositing and amplification events. Our method solves the text-to-image synthesis problem with a unique sketch optimization technique. The first stage-I GAN captures simple colors and information of objects based on annotations. The next stage, Stage-II GAN, improves Stage-I results by correcting imperfections and adding additional details, resulting in higher resolution and better image quality.

Various qualitative tests are conducted to evaluate the effectiveness of our plan. The results demonstrate the effectiveness of StackGAN in generating higher resolution images (eg 256×256) with accurate and varied content. Our method outperforms existing text-to-image generator models in terms of output quality and relevance to a particular description. The success of StackGAN opens up new possibilities for many practical applications such as image processing and computer-aided design. By accurately translating the annotation into high-quality images, our method helps improve computer vision performance and expands the capabilities of image synthesis.

REFERENCES

[1] Scene Retrieval With Attentional Text-to-Image Generative Adversarial Network by rintaro yanagi,ren togo ,takahiro ogawa, & miki haseyama.

[2] Text to Image Synthesis for Improved Image Captioning by md. zakir hossain, ferdous sohel, mohd fairuz shiratuddin,hamid laga, mohammed bennamoun.

[3] Semantic Object Accuracy for Generative Text-to-Image Synthesis by Tobias Hinz , Stefan Heinrich , and Stefan Wermter.

[4] Dynamic Memory Generative Adversarial Networks for Text-to Image Synthesis by Minfeng Zhu1,Pingbo Pan Wei Chen Yi Yang.

[5] Real-world Image Construction from Provided Text through Perceptual Understanding by Kanish Garg, Ajeet kumar Singh, Dorien Herremans and Brejesh Lall.

[6] Research on Text to Image Based on Generative Adversarial Network by Li Xiaolin1, Gao Yuwei.

[7] Super-Resolution based Recognition with GAN for Low-Resolved Text Images by Ming-Chao Xu,Fei Yin, and Cheng-Lin Liu.

[8] A Novel Visual Representation on Text Using Diverse Conditional GAN for Visual Recognition by Tao Hu , Chengjiang Long, Member, and Chunxia Xiao , Member.

[9] A Simple and Effective Baseline for Text-to-Image Synthesis by Ming Tao Hao Tang Fei Wu1 Xiaoyuan Jing Bing-Kun Bao Changsheng.

[10] Knowledge-Transfer Generative Adversarial Network for Text-to Image Synthesis by Hongchen Tan , Xiuping Liu , Meng Liu , Baocai Yin, and Xin Li.

[11] Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I. J., Bergeron, A., Bouchard, N., and Bengio, Y. (2012). Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop.

[12] Bengio, Y. (2009). Learning deep architectures for AI. Now Publishers.

[13] Bengio, Y., Mesnil, G., Dauphin, Y., and Rifai, S. (2013). Better mixing via deep representations. In ICML'13.

[14] Bengio, Y., Thibodeau-Laufer, E., and Yosinski, J. (2014a). Deep generative stochastic networks trainable by backprop. In ICML'14.

[15] Bengio, Y., Thibodeau-Laufer, E., Alain, G., and Yosinski, J. (2014b). Deep generative stochastic net works trainable by backprop. In Proceedings of the 30th International Conference on Machine Learning (ICML'14).

[16] Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., and Bengio, Y. (2010). Theano: a CPU and GPU math expression compiler. In Proceedings of the Python for Scientific Computing Conference (SciPy). Oral Presentation.