TECHNICAL DOCUMENT 3418
May 2022

TextCycleGAN FY20

Mohammad R. Alam
Nicole A. Isoda
Mitch C. Manzanares
Anthony C. Delgado
Antonius F. Panggabean

NIWC Pacific

Naval Information Warfare Center Pacific (NIWC Pacific)
San Diego, CA 92152-5001

This page is intentionally blank.

TECHNICAL DOCUMENT 3418
May 2022

TextCycleGAN FY20

Mohammad R. Alam
Nicole A. Isoda
Mitch C. Manzanares
Anthony C. Delgado
Antonius F. Panggabean

NIWC Pacific

Administrative Notes:
This technical document was approved through the Release of Scientific and Technical Information (RSTI) process in October 2020 and formally published in the Defense Technical Information Center (DTIC) in May 2022.

NIWC Pacific
San Diego, CA 92152-5001

NIWC Pacific
San Diego, California 92152-5001

A. D. Gainer, CAPT, USN
Commanding Officer

W. R. Bonwit
Executive Director

ADMINISTRATIVE INFORMATION

The work described in this report was performed by the Intelligent Sensing Branch of the Basic and Applied Research Division, Naval Information Warfare Center Pacific (NIWC Pacific), San Diego, CA. The NIWC Pacific In-House Laboratory Independent Research (ILIR) Program sponsored by the Office of Naval Research (ONR) provided funding for this Basic Applied Research project.

Released by
Ayax Ramirez, Division Head
Basic and Applied Research Division

Under authority of
Carly Jackson, Department Head
Cyber/Science & Technology Department

Editor: RJP

EXECUTIVE SUMMARY

OBJECTIVE

TextCycleGAN (TCG) is a new image captioning framework on a cyclical generative adversarial network (CycleGAN) foundation. This effort seeks to explore the performance of various CycleGAN and conditional GAN architectures to construct the TCG image captioning software package.

METHODS

The development TCG proceeded as follows:

- Research and replication of state-of-the-art (SOTA) GAN architectures, alternative image captioning frameworks, and candidate CycleGANs and conditional GANS to set the foundation and direction of the work (performed in FY19)
- With candidate architectures chosen in FY19, TCG's focus on implementing a full CycleGAN capable of generating descriptions of imagery provided, or captions, and imagery from captions

CONCLUSIONS AND RECOMMENDATIONS

GAN replication and the testing of candidate architectures yielded promising results. Image synthesis conditional GANs, StackGAN++ and the text-to-image-to-text architecture, were tested and performed well in image generation; however, image caption GANs required more time for testing and replication. Although the test-to-image-to-text architecture was based on a CycleGAN, it was missing a few key components such as full text generation and a cycle-consistency loss. As a result, multiple GAN architectures have been replicated and verified, but the full TCG architecture has yet to be constructed due further testing of approaches required for the image captioning portion as well as needed development of a cycle-consistency loss.

This page is intentionally blank.

CONTENTS

FIGURES

This page is intentionally blank.

1. INTRODUCTION

In [1] we introduced TextCycleGAN (TCG): an image captioning framework based on cycle-consistent gen- erative adversarial networks (CycleGANs). A robust image captioning framework can provide benefit to image search and information retrieval by providing automatic and detailed descriptions of imagery. This report discusses changes made to the original design of TCG architecture from [1]. The rest of this paper is organized as follows. Section 2 prior work related to TCG and key concepts behind the methods used in this updated implementation. Section 3 presents the updated methods for TCG and reasoning for these changes to the architecture. Section 4 discuss the current status of development on the project. Concluding remarks can be found in Section 5.

This page is intentionally blank.

2. BACKGROUND

2.1 IMAGE SYNTHESIS

Although we reviewed related work in the domain of image captioning in [1], we did not review prior work in the realm of image synthesis. Over the last few years, GANs have shown promising results in generating high quality images from text descriptions. [2] successfully uses text descriptions to condition their model in order to generate plausible images. They also introduced a manifold interpolation in order to improve the quality of the generated images. Following their work, [3] introduce StackGAN. StackGAN decomposes the problem of generating high resolution images into sub-problems. At each stage it generates a higher resolution image by conditioning the model on the previous stage's output and the text description. They also introduce a Conditioning Augmentation technique to stabilize the training process. In [4] the authors of [3] build off their previous work and introduce StackGAN++. StackGAN++ is similar to StackGAN in that it is a multi-stage GAN, but StackGAN++ can handle both the conditional and unconditional generative tasks. It consists of multiple generators and discriminators in a tree structure and generates images of different sizes at each stage. It stabilizes its training process by jointly approximating multiple distributions.

2.2 SEARCH METHODS

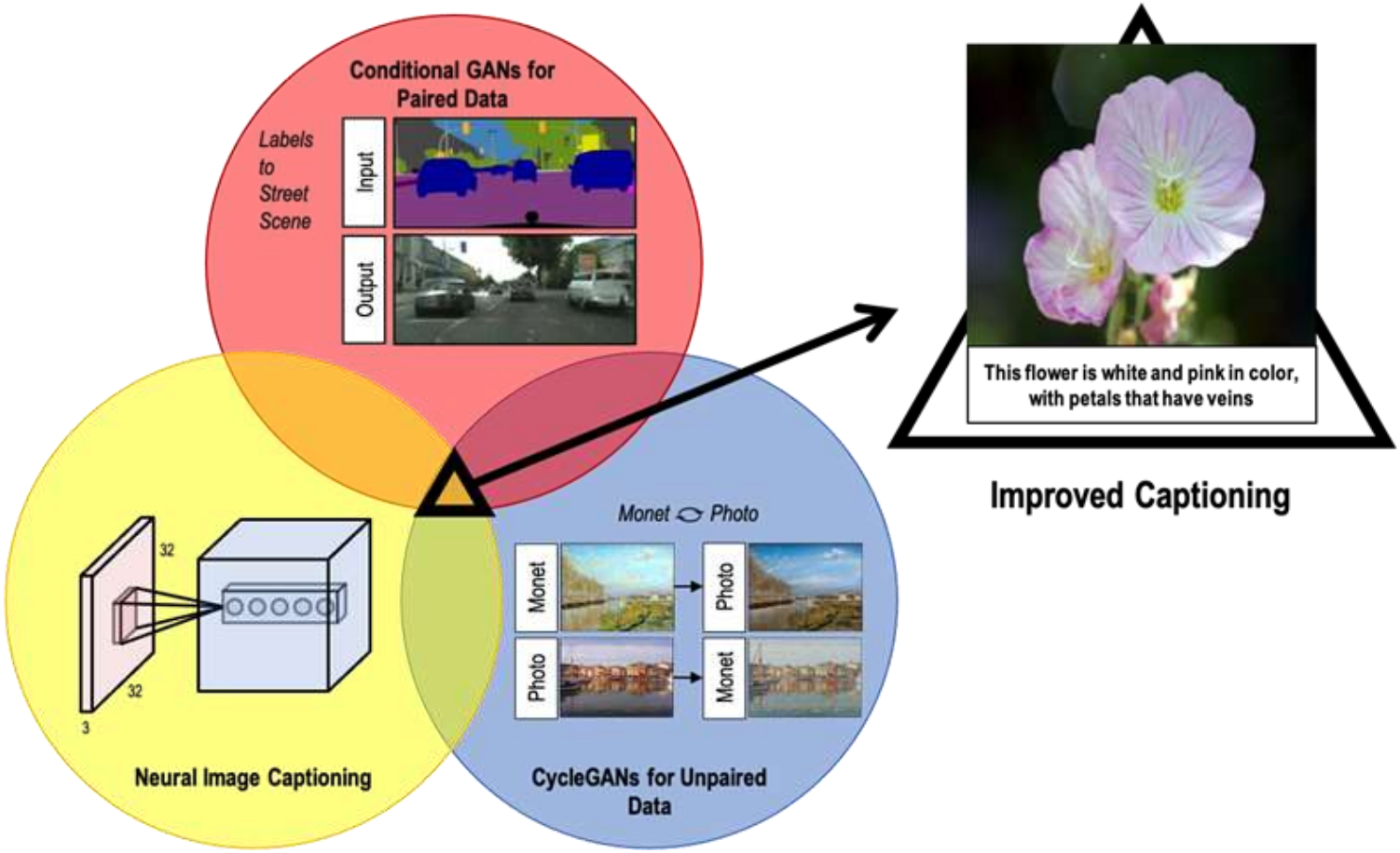
In natural language processing (NLP), image captioning, machine translation, and text summarization models generate probability distributions across a vocabulary of output words and use a decoding algorithm to generate the most likely output sequences of words from the probability distribution. Getting the most likely output sequences means searching through all the possible output sequences based on their probabilities at each time step. In practice, this search is done using heuristic search methods such as Greedy Search or Beam Search.

2.2.1 Greedy Search

Greedy Search is a simple search algorithm that selects the word with the highest probability at each time step as the sequence is constructed. The search continues until a maximum sequence length is reached or an end-of-sequence token is reached. The main benefits of this approach are that it is very fast and does not use a lot of memory. However, the quality of the output sequence is not always optimal.

2.2.2 Beam Search

Unlike Greedy Search, as the sequence is constructed the Beam Search algorithm expands all the possible next words and then keeps the top K sequences based on conditional probability. K is a tunable parameter known as Beam Size, and it corresponds with the number of best sequences to keep at each time step. The main benefit of this approach is that it considers multiple best options which results in higher quality output sequences. Beam Search allows for a good balance between computational overhead and search quality by using the tunable parameter K. It should be noted that by using K = 1, Beam Search can be treated as a Greedy Search.



- This image serves to highlight the inspiration and foundation for the methods used in developing TextCycleGAN and demonstrate its capability in improving image captioning.

Figure 1. Comparative overlap leading to improved captioning.

3. METHODS

3.1 WORD AND SENTENCE EMBEDDINGS

Sentence embeddings are used for caption comparison for the discriminator. As captions are sampled from the generator, they are run through a sentence embedding network and their distance is compared with both the sample image embedding and all prior sample sentence embeddings. For our purposes, we used the skip-thought sentence embedding model. The skip-thought model, first created by [5], encodes a given sentence into a vector and then uses a decoder to generate sentences that would come before and after the input sentence contextually. This framework simplifies the process of comparing distance between sample sentences.

-

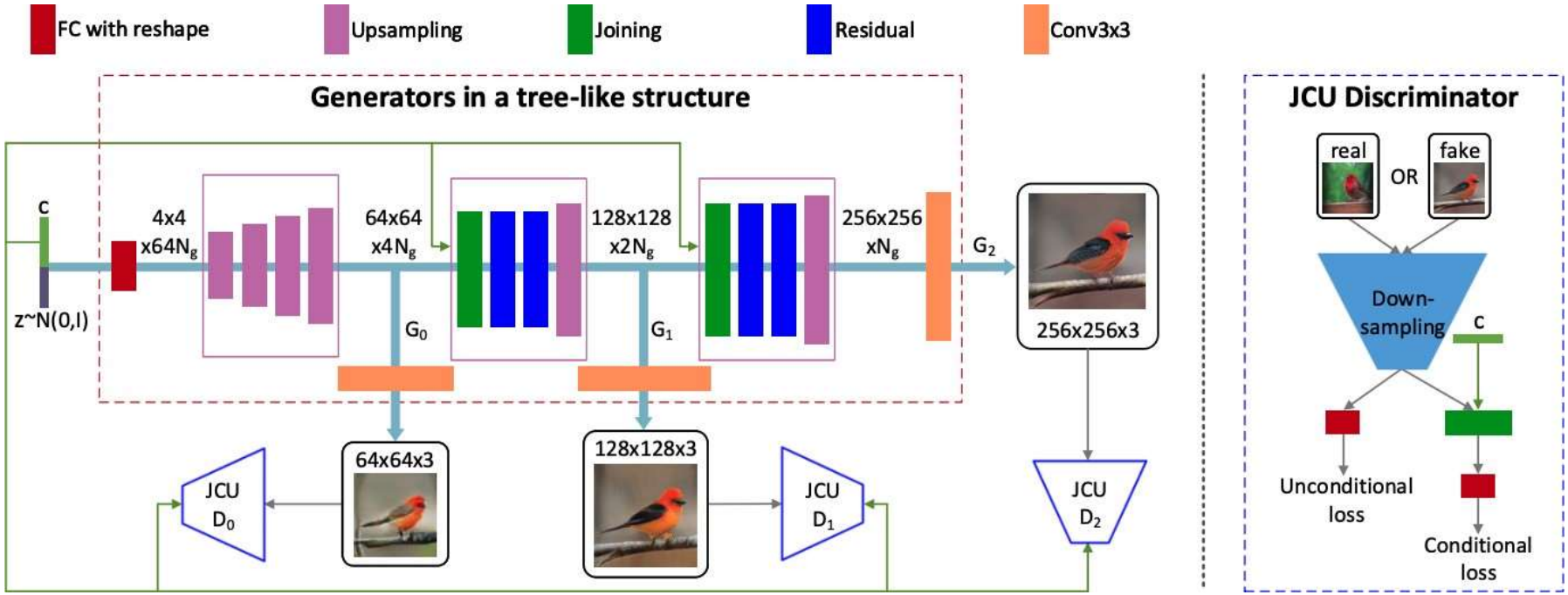


Figure 2. StackGAN++ framework as described in [4].

3.1.1 Image Captioning

We considered the image captioning architectures from [6] and [7]. [6] uses a convolutional neural network (CNN) to extract features to provide as attention/context at each step of their single-layered LSTM. [7] uses CNN features in conjunction with explicit object detection features as initial context, then continues using just the CNN features as context for each step of their multi-layered LSTM. Additionally, [7] uses a Gumbel Sampler to select an output at each step of the LSTM. We opted to follow the simpler, attentional approach of [6], but maintained the multi-layer LSTM with Gumbel Softmax approach of [7]. Furthermore, a multi-layer LSTM had the potential to reduce training time/resources needed at any given time due to their typically smaller size per layer compared to a single-layered LSTM architecture (this was necessary to address hardware limitations on our end).

3.1.2 Search methods

For TextCycleGAN (TCG), we opted to use Beam Search as our search method because we are focused on improving the quality of the captions that are associated with each image. Using the tunable parameter K, we can control how many sequences the algorithm considers at each time step and have a greater chance of a high-quality output sequence. We also have the added benefit of greater control over the resources used by the algorithm since we can tune the K to reflect the resources available to train with, allowing us to use our resources more efficiently.

3.1.3 Gumbel-Softmax

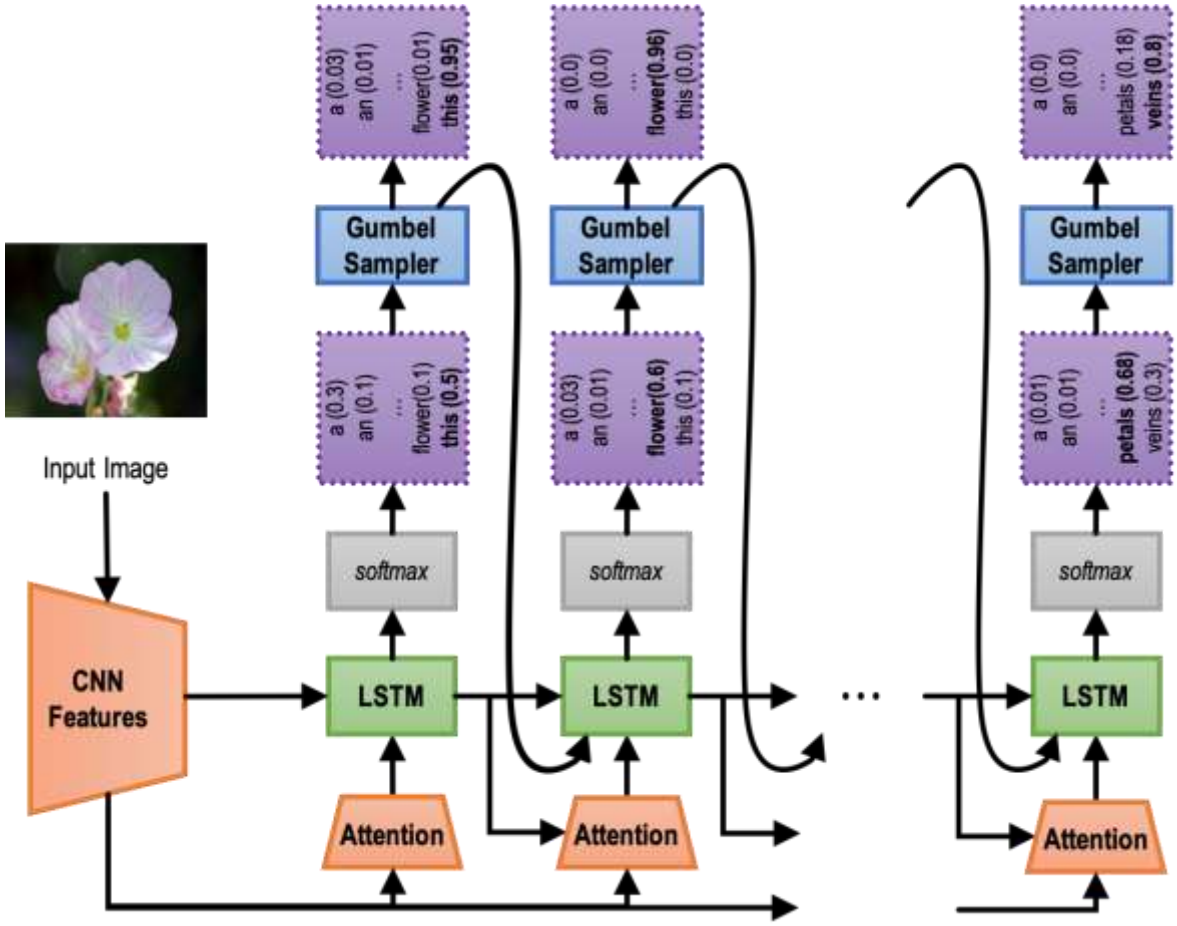
The Gumbel-Softmax distribution was discovered independently by [8, 9] as a way to enable gradient esti- mation for categorical, non-differentiable samples. Beginning with a random variable r from a categorical distribution parameterized by $\Theta = \theta_0, ..., \theta_{v-1}$, r can be expressed as

$$r = \text{one\_hot}\left[\arg\max_i (g_i + \log\theta_i)\right], \tag{1}$$

with *g's i.i.d.* from a standard Gumbel distribution. To do a continuous relaxation of r, replace the argmax with softmax:

$$r' = \text{softmax}\left[\frac{g_i + \log\theta_i}{\tau}\right], \tag{2}$$

where τ is a temperature parameter. As τapproaches zero, the distribution becomes closer to one-hot, e.g. $r^l = r$ when $\tau = 0$. [7] utilized Gumbel sampling to improve diversity and naturalness of generated captions. We applied a similar technique to improve the captions in our experiments to better match natural human language. We apply Gumbel-Softmax to the logits in each layer of our LSTM and pass this (now) differentiable estimation to the following layer.



Convolutional features are input to the LSTM to generate a sentence. A Gumbel Sampler obtains soft samples from the softmax, thus allowing backpropagation.

Figure 3. Image captioning model as inspired by [6] and [7].

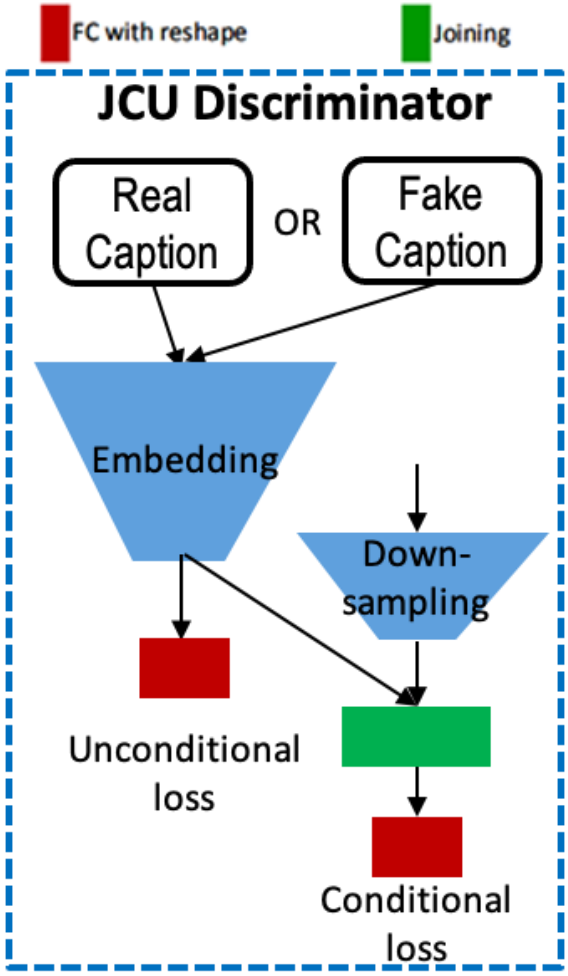
3.1.4 Caption Discriminators

At the discriminator input, an embedded image is joined with embedded captions to determine fake or real. The same joint conditional and unconditional discriminators utilized in [4] for image synthesis are applied to image captioning to approximate caption distributions. The conditional discriminator consists of joint convolutions of the caption and image matching with sigmoid and logits applied. The unconditional discriminator is simply a fully connected layer with a sigmoid activation.

The discriminator outputs both the conditional and unconditional results from the set of captions. This allows a determination based on both authenticity and fit with conditional inputs. In an experiment in [4] this results in a higher inception score than conventional discriminators.

3.2 IMAGE SYNTHESIS

The image synthesis architecture we chose is based on StackGAN++ from [4]. StackGAN++ breaks the problem of generating images into manageable sub-problems by utilizing multiple generators and discrimina- tors in stages to generate images up-sampled with unconditional and conditional distribution approximations. Each stage of StackGAN++ generates a higher resolution image by combining the previous stages output and the condition variable. The original StackGAN++ is used to create images of size 256 x 256 by starting with generating an image of size 64x64 then upscaling the image to 128 x 128 and finally 256 x 256 at each new stage. We opt to stop the upscaling at 128 x 128 for TextCycleGAN because our main focus is on the image captioning and not the image synthesis. This allows us to save resources and speed up training.



-

The joint-conditional and unconditional discriminator is shown above. This joint loss drives caption generation through a combined assessment of the caption's authenticity with respect to its input image and without it..

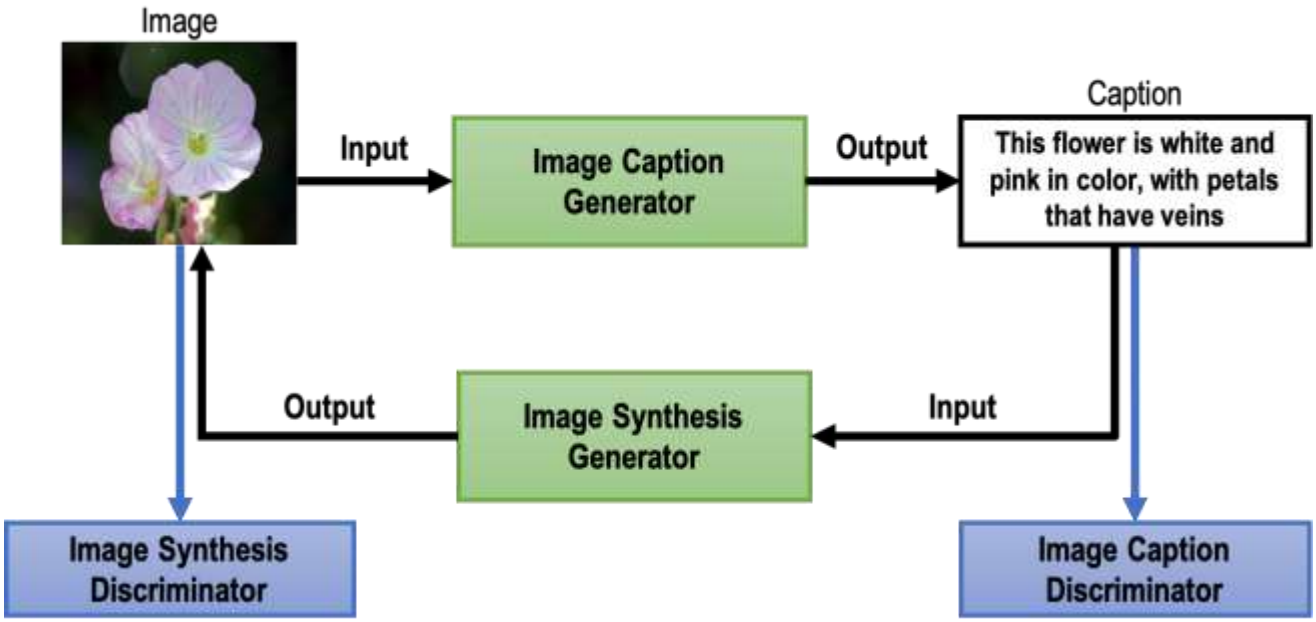
Figure 4. The JCU discriminator.

We chose to use StackGAN++ because it gave us control over not only the image size but also the quality. StackGAN++ jointly approximates multiple related distributions such as conditional and unconditional dis- tributions. The unconditional loss determines if the image is real or fake and the conditional loss determines if the condition matches the input. This creates a more stable training structure allowing for higher quality images to be generated. Finally, StackGAN++ adds a color-consistency regularization to keep the sample images generated more consistent in color which improves the overall quality of the images.

Although the images generated are not the focus of the entire TextCycleGAN system, being able to generate high quality images is important for the overall success of the system. This is because of the need to map a sample from one domain to another then being able to translate that back in CycleGANs. The more realistic the images we can generate then the better the captions for those images can become.

3.3 CYCLE CONSISTENCY

An important property of CycleGANs is maintaining cycle-consistency [CycleGAN paper]. Specifically, if we define both GANs of a CycleGAN as functions A and B, an input that is passed through both A then B, or B then A, should be recreated. This behavior of recreating the original input is made possible through cycle-consistent loss functions. This is a well-defined problem for one-to-one transforms like that of the image transforms in [CycleGAN Paper], such as horses-to-zebras or winter-to-summer. For TCG, perfect or near- perfect recreation would mean ignoring the inherent many-to-many problem associated with image captioning and image synthesis. Ensuring true cycle-consistency between the original input and recreated output in our case is too constrained. To address this, we enforced cycle-consistency in the embeddings to ensure variability in the final output while maintaining similarity in the objects' feature space. We accomplish this by performing an L1 loss between the recreated input and output embeddings for both images and captions. For imagery, we utilize [StackGAN++]'s feature encoding downsampler in the discriminator to encode the imagery and then perform the L1 loss on these encoded images. For captions, we simply take the sentence embeddings (using [skip-thoughts]) for both the recreated and input captions and perform the L1 loss between the two. This helps maintain consistency in the feature space while allowing variability in the output space.



- The goal is to utilize cycle-consistency on sentence embeddings and image features where function A is the image captioning process and function B is the image generation process.

Figure 5. High-level TCG architecture.

This page is intentionally blank.

4. CURRENT STATUS

As of this writing, the project is in a testing and debugging phase. We are conducting iterative training and testing of our final TextCycleGAN architecture on publicly available datasets. We are working with limited computing power and shared resources. In its current state, the model is too resource-intensive and requires greater than the amount of GPU memory we have available to us. Changes to model parameters can reduce resource requirements at the cost of performance. Additionally, prior tests with beam search showed that the algorithm would take longer than a day to complete one epoch. Reducing the queue size of the beam search method or using greedy search would reduce training time, but also decrease quality of output sentences. Utilizing CPU and GPU parallelization can also help by distributing processing across multiple devices. We are also looking into using other computing resources to expedite training and testing.

This page is intentionally blank.

5. CONCLUSION AND FUTURE WORK

With TCG's core implementation finished, all that remains is to continually train and test the model to analyze its performance and make modifications as necessary. Once testing has finished, we will be moving from applying our model on publicly available datasets to other curated datasets. As discussed in Section 4, we are currently stalled by limited computing power hindering training and model performance issues that making training slow. We plan on making changes to our model parameters to increase training speed and utilizing additional machines to increase computing capability. Once we are able to smoothly train and test our model, we will be able to easily compare the performance of our model to the other image captioning frameworks and make additional adjustments as necessary.

This page is intentionally blank.

REFERENCES

1. Mohammad Alam, Iryna Dzieciuch, Maurice Ayache, Nicole Isoda, and Mitch Manzanares and Anthony Delgado. 2019 “Textcyclegan F19 technical report”.
2. Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016 “Gen- erative adversarial text to image synthesis”.
3. Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan. 2016 “Text to photo-realistic image synthesis with stacked generative adversarial networks”.
4. Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan. 2017 “Realistic image synthesis with stacked generative adversarial networks”.
5. Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015 “Skip-thought vectors”.
6. Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell. 2015 “Neural image caption generation with visual attention”.
7. Rakshith Shetty, Marcus Rohrbach, Lisa Anne Hendricks, Mario Fritz, and Bernt Schiele. “Speaking the same language: Matching machine to human captions by adversarial training”. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. doi: 10.1109/iccv.2017.445. URL http://dx.doi.org/10.1109/ICCV.2017.445.
8. Eric Jang, Shixiang Gu, and Ben Poole. 2016 “Categorical reparameterization with gumbel-softmax”.
9. Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2016 “The concrete distribution: A continuous relaxation of discrete random variables”.

This page is intentionally blank.

INITIAL DISTRIBUTION

84310	Technical Library/Archives	(1)
71740	M. R. Alam	(1)
71740	N. A. Isoda	(1)
71740	M. C. Manzanares	(1)
71740	A. C. Delgado	(1)
71740	A. F. Panggabean	(1)

Defense Technical Information Center
Fort Belvoir, VA 22060–6218 (1)

This page is intentionally blank.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-01-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
May 2022	Final	

4. TITLE AND SUBTITLE

TEXTCYCLEGAN FY20.

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHORS

Mohammad R. Alam
Nicole A. Isoda
Mitch C. Manzanares
Anthony C. Delgado
Antonius F. Panggabean
NIWC Pacific

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

NIWC Pacific
53560 Hull Street
San Diego, CA 92152–5001

8. PERFORMING ORGANIZATION REPORT NUMBER

TD 3418

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
8500 Bluffstone Cove, Ste A201
Austin, TX 78759

10. SPONSOR/MONITOR'S ACRONYM(S)

ONR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A: Approved for public release. Distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In this report, we discuss improvements to TextCycleGAN: a cycle-consistent generative adversarial network (CycleGAN) for image captioning. CycleGANs train separate Generative Adversarial Networks (GANs) to learn mappings between multiple domains and strengthens each individual mapping with cycle consistency loss. As such, with CycleGANs we can create a better image captioning generator by jointly training an image synthesis generator. Since cycle-consistency ensures minimal change with recreation of the input, this offers a unique challenge for image captioning due to the many-to-many nature of the mapping from images to captions and vice-versa. We will further discuss how we tackle this many-to-many challenge as well as both image captioning and image synthesis in the report.

15. SUBJECT TERMS

Image syntheses; TextCycleGAN; CycleGAN; image captioning generator; image synthesis generator; image captioning; GAN; computer vision; natural language processing

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Muhammad R. Alam
U	U	U	SAR	32	19B. TELEPHONE NUMBER *(Include area code)* 619-940-3360

This page is intentionally blank.

This page is intentionally blank.

Naval Information Warfare Center Pacific (NIWC Pacific)
San Diego, CA 92152-5001