

Avian Artistry: Creating Images of Birds from Textual Descriptions

Akshara Nair
MT22008
Dept. of CSE, IITD
akshara22008@iiitd.ac.in

Aryan Chaudhary
MT22019
Dept. of CSE, IITD
aryan22019@iiitd.ac.in

Medha
MT22110
Dept. of CSE, IITD
medha22110@iiitd.ac.in

1 Introduction

The proposed problem of the project is to generate high-quality images which preserve the semantic meaning of the textual descriptions. The project aims to enhance the training procedure by exploring modifications to the architecture and addressing the issues of training instability, inefficiency, and loss of intra-channel relationships in traditional GANs. By using QGANs, this project aims to reduce the number of parameters needed to generate high-quality images, thus improving the model's efficiency. Additionally, the project investigates the use of spectral normalization as a regularisation technique to enhance the stability and performance of QGANs,

2 Related Work

2.1 Generative Adversarial Nets (GANs) and Quaternion GANs

Generative Adversarial Nets (GANs) is a deep learning framework that consists of two neural networks termed as Generator and Discriminator (Goodfellow et al., 2020). In the process of training, the generator network learns to generate new data samples, inhibited with noise to produce indistinguishable samples from the real images while the discriminator network learns to distinguish between real and generated samples.

Quaternion Generative Adversarial Nets (QGANs) (Grassucci et al., 2022) are a variant of Generative Adversarial Networks (GANs) that use quaternions, a four-dimensional hypercomplex number system, instead of real or complex numbers. The quaternion space allows for more efficient computation of certain operations, such as rotations, compared to the traditional real or complex number space.

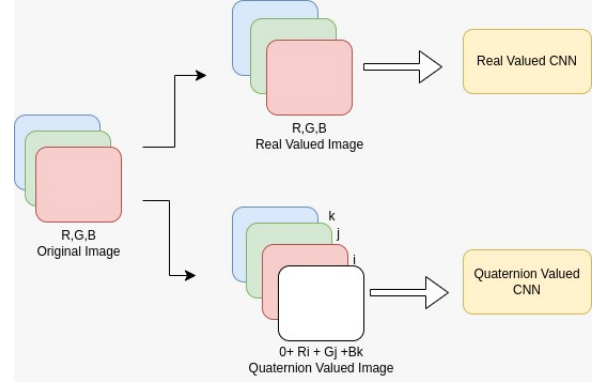


Figure 1: The proposed novel architecture utilizes the concept of Quaternion for Image generation as shown in Fig. 1 here.(Grassucci et al., 2022)

2.2 Image Generation through Captions

One of the major research leading to Text-based Image Synthesis is development of a generative model that can automatically generate images based on a given textual description or caption (Mansimov et al., 2015). The model uses a novel attention mechanism that allows it to focus and efficiently capture fine-grained details of different parts of the textual input as it generates different parts of the image to produce more realistic images.

3 Dataset Description

The CUB (Caltech-UCSD Birds-200-2011) (Wah et al., 2011) dataset is an extensively used dataset containing 11,788 images of 200 bird species, with an average of 60 images per species. The images were collected from online resources and annotated with detailed textual descriptions of the bird species.

4 Methodology and Experimental Setup

The details for Baseline-1 and Baseline-2 are as discussed below. Both codes are developed using PyTorch library.

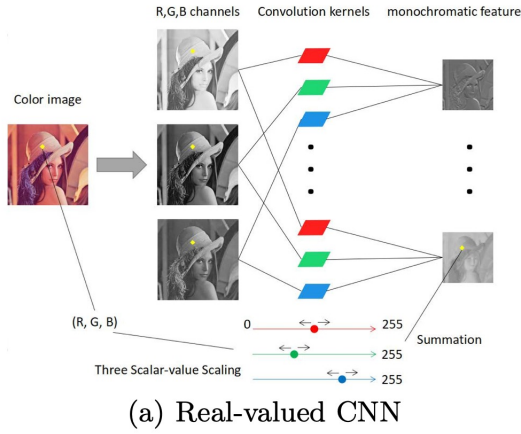


Figure 2: The diagram in Fig.-2 and Fig-3. as referenced from (Zhu et al., 2018), is describing the intermediate process of learning for the case of CNN

4.1 Baseline-1

Baseline 1 was reproduced from existing GAN architecture. The Hd5 file taxonomy is used which consists of the attributes as 'Name', 'img', 'embeddings', 'class' and 'txt' to re-train the model along with the standard datasets to load corresponding dataset. The Batch Size is set to 256. The loss functions used are as BCE, L1 and MSE for calculating Adversarial Loss, L1 Loss and L2 Loss respectively. The model is trained for approximately 200 Epochs and the embeddings are generated from Distilled BERT for inference.

4.2 Baseline-2

For second baseline, the StackGAN model architecture consisting of two stages with a generator and discriminator in each is used. The Batch Size is 24 and Optimizers used are Adam. The Learning Rate for the Generator and the Discriminator is 0.002. The Model is trained for 2,10,000 epochs and the text embedding used are generated from CNN-RNN. The model is not trained and the output is directly reproduced from the pre-trained model and the learnt model weights.

5 Proposed Novelty

While the baseline GAN models have achieved impressive results in generating high-quality samples, they often require huge models with a lot of parameters making it computationally expensive and less accessible. Also, traditional CNNs have limitations when dealing with multi-channel inputs like colour

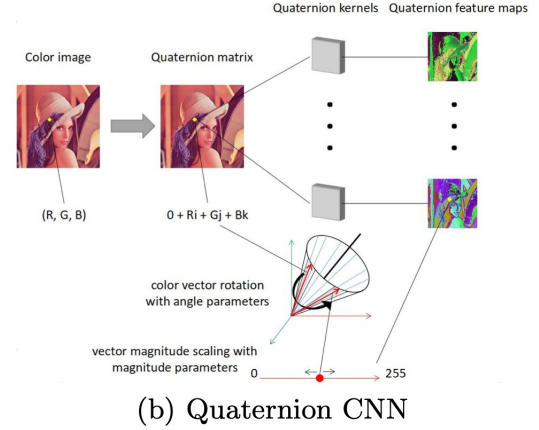


Figure 3: The diagram Fig.-2 and Fig-3. as referenced from (Zhu et al., 2018), is descriptive of the intermediate process of learning for the case of Quaternion CNNs.

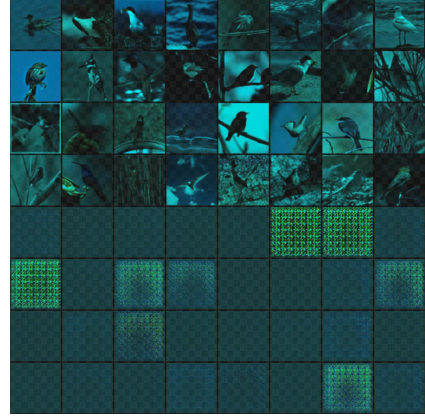


Figure 4: The last 32 images are as generated by the generator for an initial epoch.

images, as the convolution kernels simply sum up the outputs of different channels, potentially losing important structural information about color and increasing the risk of overfitting. Additionally, training process for GANs are unstable because of its dual player nature leading to the requirement of techniques to stabilise the training process. To address these problems, this project proposes three novel approaches for creating images of birds from textual description.

5.1 Quaternion GANs(A2)

The architecture of quaternionic GANs is similar to that of traditional GANs, with some key differences in how the data is represented and processed. The generator network generates a quaternionic output, which is a four-dimensional representation of an image or other data, from a random vector or noise

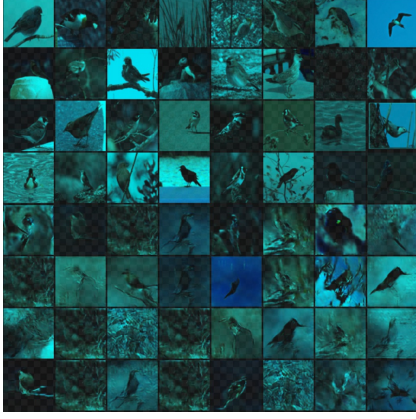


Figure 5: The last 32 images are as generated by the generator for final epochs.

input. The network consists of several layers of quaternion fully connected or quaternion transpose convolutional neural networks. The discriminator network distinguishes between real and fake samples with a quaternionic input and consists of several layers of quaternion fully connected or quaternion convolutional neural networks. Quaternionic numbers represent four-dimensional complex data, which may be more expressive than traditional two-dimensional complex numbers. This can result in more accurate and efficient image representations. Due to the Hamilton product, Quaternionic GANs have restricted degrees of freedom, making them more stable during training than traditional GANs. Quaternionic GANs reduce the number of parameters by 75 % while providing comparable or better results.

5.2 Spectral Normalised GANs(A3)

Spectral Normalization is a technique applied to GANs to enhance the stability of the discriminator during the training process. It normalizes the spectral norm of the discriminator's weight matrices to reduce the likelihood of gradient update imbalance between the generator and discriminator. Spectral Normalization enhances the smoothness of the discriminator's decision boundary and decreases the risk of mode collapse during the training process. However, it also reduces the capacity of the discriminator to differentiate between real and fake samples and increases computation complexity due to the need for calculating the Singular Value Decomposition of the weight matrix.

5.3 Quaternion SNGANs(A4)

Quaternionic spectral normalization is a method of spectral normalization that is developed for quaternionic neural networks, which use quaternions instead of real numbers to represent data and parameters. SVD calculations, orthogonalisation and normalisation are adapted according to the Quaternion domain. Just like Spectral normalisation here also Quaternionic Spectral normalisation GANs suffer loss in capacity of the discriminator, training overhead and hyperparameter sensitivity.

6 Result

The results produced by training and the inference from proposed model are as in 4 and Fig. 5. Initially, the images generated by the generator are noisy and hence do not pose a problem for the discriminator to distinguish between the real and fake images generated by the generator.

6.1 Evaluation Metrics

Hash Functions are extremely efficient for detecting the extent of similarity between images as they are extremely robust against minor distinctness. The goal is to generate Hash Functions for the original image and for the target image and take the difference among the generated hash values. If the average value of the loss over the set of all the images generated is relatively lower, the architecture is considered to generate rather realistic images. In the following sub-section, the procedure for evaluating the Hash Value is described. Another evaluation metric specified here is as **Frechet Inception Distance (FID) Score**

6.1.1 Average Hashing

It deploys the technique to calculate the mean intensity of the pixel values of the image generated and then compares each of the pixel values to the evaluated mean. If the pixel intensity is higher than the set value of the mean, the value of the pixel is set to 1, and else is set to 0. This metric is abbreviated as M-1 as in Table-1.

6.1.2 Perceptual Hashing or Discrete Cosine Transform Hashing

The hash values are evaluated by using Discrete Cosine Transform. The digital fingerprint is originally used to identify the intensity of alteration of distortion introduced in the generated image as and when compared to the original image. The High-frequency components consist of information about

the texture of the image and are rather sensitive to distinct images than low-frequency components. If the value of a high-frequency component is greater than the threshold value, it is set to 1, otherwise, it is set to 0. The resulting binary sequence consists of two values 0 and 1, representing the image finally. This metric is abbreviated as M-2 as in Table-1.

6.1.3 Difference Hashing

As the name indicates, the procedure to generate the hash values involves using a difference among the pixel intensity values of consecutive neighboring pixels of the image. Note that for this hash function specifically, the output generated is a binary string that consists of binary values, indicative of whether the current pixel value is lighter or darker as compared to the pixel next to it. This metric is abbreviated as M-3 as in Table-1.

Models	M-1	M-2	M-3
A-1	31.265	31.316	31.661
A-2	31.858	31.349	31.742
(A-3)	32.425	31.228	31.957
A-4)	31.438	31.257	31.589

Table 1: Performance of the proposed novelties as compared to the Baseline on the basis of Hash Function Based Evaluation Metrics.

Models	M-4	M-5
(A-1)	132.465	177.490
(A-2)	220.544	114.094
(A-3)	230.650	275.3
(A-4)	352.262	323.598

Table 2: Performance of the proposed novelties as compared to the Baseline on the basis of FID-Score.

6.1.4 Fréchet Inception Distance (FID)

The FID Score calculates the distance between the multivariate Gaussian distributions of the feature vectors obtained from the reference dataset and the generated images. A lower value of the FID Score implies a better performance of GAN. The formula to evaluate the FID-Score is as: The FID score is evaluated using the formula as in Equation (1).

$$\text{FID} = \|\mu_1 - \mu_2\|^2 + \text{Tr}(\Sigma_1 + \Sigma_2 - 2(\Sigma_1 \Sigma_2)^{1/2}) \quad (1)$$

where μ_1 and μ_2 are the mean feature vectors of the reference dataset and the generated images, respectively, and Σ_1 and Σ_2 are the covariance matrices of the reference dataset and the generated images, respectively. This metric is abbreviated as M-4 and M-5 in Table-1. M-4 typically specifies the evaluation of loss with the validation set which consists of RGB images and the images in the Validation Set are also transformed to a 4-Channel image by introducing another channel as 'alpha'. Now both the set of images are comparable and hence, when the FID-Score is calculated for the case of M-5, there is observed a significant drop in the value of FID-Score, which strongly supports the efficiency of the proposed novel architecture over Baseline.

6.2 Observation of Results

There is a significant improvement in the FID-Score, after modifying the architecture to use Quaternion-based architecture. Other experimentations to stabilize the training of discriminator, included the introduction of Spectral Normalization and combining Quaternion with Spectral Normalization. Both of these experimentations, led to unsatisfactory results as there was no improvement in the FID-Score. However, the quality of the images being generated is not compromised. This can be inferred by observing the evaluated value of similarity of Hash Values calculated, which is also averaged over the entire dataset.

6.3 Error Analysis

The following table as 3 and 4 summarises the results of where the proposed architecture is highly probable to generate irrelevant images.

7 Conclusion

By implementing the discussed architecture, the Discriminator Network which consisted of total of 2,800,064 parameters was transformed to the Quaternion Discriminator Network, with a reduced count of total parameters as 709,376. For the case of Generator Network, the total number of parameters is 4,133,504 and the Quaternion Generator Network has a reduced count of the total parameters as 1,035,200. Also, we observed improvement in FID scores over the baseline model GANS.

8 Contributions

This project was a collaborative effort between team members who each made significant con-




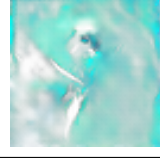

	1	2	3	4	5
Text	Text - A brilliantly orange colored bird with black head, nape and tail, and black wings has white wing bars.	Text - This small bird has a grey bill and crown and grey wings with white wing bars	This bird has a medium beak with mostly yellow feathers.	This small bird has a light brown breast and belly and a small belly-pointed beak.	This bird has a large beak and a long neck.
Image					

Table 3: Table with Images which are generated according to the intent of the user in the input prompt.





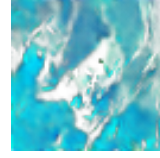
	6	7	8	9	10
Text	Text- This bird has wings that are grey and has a white belly.	Text - A small bird with a black head and yellow underbelly.	This bird is brown black in color with a light pink beak, and brown eye rings.	This bird has wings that are brown and has a yellow belly.	This bird is grey with white and has a very short beak.
Image					

Table 4: Table with Images which are generated contrary to the the intent of the user in the input prompt.

tributions. Aryan designed the novel approach to generative modeling using quaternion algebra. Medha and Akshara extended the project by adding spectral normalization to GAN and QGANs to experiment with potential improvements in GAN training. Aryan, Medha, Akshara conducted experiments to demonstrate the effectiveness of QGANs, SNGANs, and QSNGANs, respectively. Additionally, all team members contributed to the writing and editing of the report to ensure clear and accurate presentation of findings.

References

- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.
- Eleonora Grassucci, Edoardo Cicero, and Danilo Comminiello. 2022. Quaternion generative adversarial networks. In *Generative Adversarial Learning: Architectures and Applications*, pages 57–86. Springer.
- Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. 2015. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*.
- C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. 2011. Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.
- Xuanyu Zhu, Yi Xu, Hongteng Xu, and Changjian Chen. 2018. Quaternion convolutional neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 631–647.