From Lyrics to Visuals: A Conditional GAN Framework for Album Cover Generation

Laura Wu
Stanford University
laurawjr@stanford.edu

Juliana Ma
Stanford University
juliamma@stanford.edu

Abstract

This project investigates the use of generative adversarial networks (GANs) for synthesizing original album cover art conditioned on lyrical content. We construct a custom multimodal dataset comprising albums released between 2005 and 2024, compiled from Wikipedia, Spotify, and Genius, to enable text-to-image learning. Our baseline model is a conditional Deep Convolutional GAN (DCGAN), which we extend with several architectural and training modifications, including CLIP-based text embeddings, CLIP-guided loss, and spectral normalization. The objective is to generate visually coherent album covers that reflect the thematic and emotional characteristics of the corresponding lyrics. We evaluate model performance using Fréchet Inception Distance (FID) and Inception Score (IS), and find that both architectural refinement and CLIP-based conditioning contribute meaningfully to generation quality. Our best-performing model combines a refined GAN architecture with CLIP embeddings and contrastive loss, achieving an FID of 131.66 and an IS of 4.79, outperforming all baseline variants across both metrics.

1. Introduction

The intersection of music and visual art offers rich creative possibilities, particularly in the context of making album cover art, the entire album's artistic concept and the artist's identity condensed in one single image. In the age of streaming and digital music browsing, the album cover can directly influence people's choice of music and have a deciding impact on the number of clicks and listeners a song gets, but making album cover art could be difficult, especially for independent or non-professional artists short on time and resources. Therefore, we want to leverage the power of deep generative models to generate album cover art that visually captures the emotion, tone, and style conveyed by the music. Our algorithm takes as inputs the title of an album, the titles of songs in the album and their corresponding lyrics. We then use a DCGAN conditioned on a text embedding combining a summary of the album's lyrics, its overall sentiment, title, and release year to output a predicted album cover. The model is trained on a dataset of real album covers that we put together and their associated textual features.

1.1. Literature Review

Generative Adversarial Networks (GANs), introduced by Goodfellow et al. (2014) [2], have become a foundational framework in image generation tasks. A GAN consists of two competing neural networks—a generator and a discriminator. The generator learns to produce plausible data samples, while the discriminator attempts to distinguish between real and generated samples. Through this adversarial process, the generator progressively improves, ultimately learning to synthesize realistic outputs that approximate the distribution of the training data.

Subsequent advancements have significantly enhanced the original GAN architecture in terms of image quality and resolution. For instance, Denton et al.[1] and Karras et al.[5] introduced models and training methods capable of generating images with higher fidelity and finer detail. Among these developments, Deep Convolutional GANs (DCGANs) have emerged as a particularly effective variant for image generation. DCGANs, proposed by Radford et al. (2015) [10], incorporate convolutional layers in the discriminator and convolutional-transpose layers in the generator, which enable the model to learn hierarchical image features more effectively than fully connected layers. This makes DCGANs well-suited for visually complex generation tasks such as album cover synthesis.

The use of GANs for album cover generation has attracted increasing research interest. Stoppa et al. (2022) [13] applied a DCGAN and a StyleGAN, as proposed by Karras et al. (2019) [7], to a dataset of approximately 80,000 album covers, demonstrating that while the model can learn to produce stylistically coherent outputs, it struggles to generate high-resolution images that capture fine-grained details. They identified a trade-off between

model scalability and visual fidelity: increasing the resolution can lead to either excessive training times or degradation in feature learning due to architectural limitations. Other researchers attempt to generate album covers conditioned on a variety of side information: Hepburn et al. (2017) [3] used an Auxiliary Classifier GAN (AC-GAN architecture) to condition covers on the genre label; Marien et al. (2022) [8] proposed a novel deep-learning framework to generate cover art guided by audio features.

In contrast to these approaches, we are particularly interested in conditioning our model on free-form textual inputs—specifically, song lyrics. While traditional conditional GANs often rely on discrete class labels, integrating free-form text presents a greater challenge. Reed et al. (2016) [11] proposed a conditional GAN that generates images of birds and flowers from natural language descriptions by conditioning a DCGAN on text embeddings produced by a hybrid character-level convolutional recurrent neural network (char-CNN-RNN). Their work illustrates that conditioning on textual features can yield high-quality, semantically relevant images.

However, lyrics differ significantly from the concise and visually grounded captions used by Reed. Lyrics are often abstract, metaphorical, and repetitive, making them more difficult to map directly onto visual representations. More recently, Contrastive Language–Image Pretraining (CLIP), introduced by Radford et al. (2021) [9], has emerged as a powerful method for learning joint representations of images and text. CLIP learns a joint image–text embedding space through contrastive pretraining on large-scale image–text pairs. Its ability to produce semantically rich, high-dimensional embeddings from free-form text makes it particularly well-suited for conditioning generative models on complex inputs such as song lyrics.

Together, these prior works provide a foundation for our approach: leveraging DCGANs for image synthesis, incorporating textual conditioning inspired by prior conditional GANs, and addressing the novel problem of translating lyrical and musical content into coherent visual outputs.

2. Dataset

To our knowledge, no publicly available dataset jointly provides album cover images alongside corresponding metadata such as titles, lyrics, and release information. To address this gap, we curated a custom multi-modal dataset spanning the years 2005-2024 by aggregating data from Wikipedia, Spotify, and Genius. The collection process was structured as a modular five-stage pipeline:

Metadata Extraction. We began by scraping the "List of YEAR albums" pages on Wikipedia, covering a 20-year span (2005–2024). These pages follow a relatively consistent tabular format, enabling us to systematically extract the artist name, album title, and genre for each entry. Special handling was implemented for years with non-standard formatting.

Title Normalization. To ensure consistent downstream querying, we performed lightweight text normalization on album and artist names. This included stripping whitespace and removing bracketed or parenthetical suffixes containing phrases such as "Deluxe Edition" or "album."

Lyrics and Cover Retrieval. For each cleaned album-artist pair, we queried the Spotify API to retrieve metadata, including the cover image URL and tracklist. Album cover images were downloaded at the highest available resolution. We then searched Genius using each song title and artist name to collect lyrics, saving each song's text individually.

Data Organization. The final dataset is organized by year. Each year is associated with a CSV index containing album-level metadata, and each album is stored in a dedicated directory containing the cover image and a collection of plain-text lyric files—one per track.

Lyrics Post-Processing. Raw lyrics obtained from Genius often contain structural tags (e.g., `[Verse 1]`, `[Chorus]`). We removed such markup and standardized the text to retain only the lyrical content. These cleaned lyrics are used for all subsequent modeling and embedding.

Limited by the total number of listed albums as well as the availability of cover art and relevant album metadata, we end up with a dataset of 12885 albums across 20 years, which we then randomly split into fixed train/validation/test sets by a ratio of 80/10/10.

3. Method

We begin with a baseline conditional Generative Adversarial Network for generating album cover art from textual metadata and incrementally introduce architectural, representational, and training enhancements to assess their impact. Our goal is to systematically evaluate how specific design choices affect both visual fidelity and semantic alignment with the input text.

This section details the components of our baseline and improved variants along three axes:

1. The choice of **text embedding** $\varphi(t)$ used to condition both the generator G and discriminator D.
2. The inclusion of a **CLIP-based contrastive loss** to explicitly encourage alignment between generated images and textual prompts.
3. The choice of **model architecture and training strategy** used to improve stability and output quality.

3.1. Baseline GAN

Our baseline follows the conditional DCGAN architecture introduced by Reed et al. [11]. The generator G receives a random noise vector $z \sim \mathcal{N}(0, I)$ and a text embedding $\varphi(t)$. The embedding is passed through a fully connected projection and concatenated with z. This latent vector is mapped to a $4 \times 4 \times 512$ feature map via a dense layer, and then progressively upsampled using transposed convolutions. The number of upsampling layers is adjusted based on the target image resolution (e.g., 5 layers for 128×128 outputs). Batch normalization and ReLU are applied after each layer, and a final $\tanh$ activation generates RGB outputs in $[-1, 1]$.

The discriminator D processes images through a stack of strided convolutional layers (with optional batch normalization), reducing the spatial resolution to 4×4. The text embedding is projected, broadcasted spatially, and concatenated channel-wise with the image feature map. A series of 1×1 and 4×4 convolutions yield a real/fake classification score. Leaky ReLU activations are used throughout. No spectral normalization is applied in this variant.

MiniLM Fusion Embedding. This approach constructs a 1152-dimensional embedding by concatenating three 384-dimensional vectors generated using the `all-MiniLM-L6-v2` model:

- A summary of the album's lyrics produced via a BART summarizer (`facebook/bart-large-cnn`).
- The top-3 emotions detected in the lyrics using a RoBERTa-based emotion classifier (`nateraw/bert-base-uncased-emotion`).
- A title+year descriptor, formatted as a short prompt.

Each component is encoded separately and their `[CLS]` tokens are concatenated to form the final album embedding $\varphi(t)$, which remains fixed throughout training.

Losses. Every model uses a generator G and discriminator D trained under a conditional GAN framework. The generator is conditioned on a text embedding $\varphi(t)$ that encodes album metadata, and the discriminator receives both an image and its associated text embedding.

- **Adversarial Objective.** We adopt the non-saturating GAN loss for both G and D, formulated as:

$$\begin{aligned}\mathcal{L}_D = & -\mathbb{E}_{x,t}[\log D(x, \varphi(t))] \\ & -\mathbb{E}_{z,t}[\log(1 - D(G(z, \varphi(t)), \varphi(t)))]\end{aligned} \tag{1}$$

$$\mathcal{L}_G = -\mathbb{E}_{z,t}[\log D(G(z, \varphi(t)), \varphi(t))] \tag{2}$$

  This drives the generator to produce images indistinguishable from real ones, conditioned on the same text prompt.

- **Reconstruction Loss (L_1).** To encourage low-level structural similarity between real and generated images, we include an L_1 loss:

$$\mathcal{L}_{L_1} = \lambda_1 \cdot \|G(z, \varphi(t)) - x\|_1 \tag{3}$$

- **Feature Matching Loss (L_2).** Additionally, we penalize differences between discriminator feature activations on real and generated images. Let $f_D(\cdot)$ denote an intermediate feature map in the discriminator; then:

$$\mathcal{L}_{L_2} = \lambda_2 \cdot \|f_D(G(z, \varphi(t))) - f_D(x)\|_2^2 \tag{4}$$

  This stabilizes training and improves perceptual quality by matching higher-level feature statistics.

The total generator loss combines all applicable components:

$$\mathcal{L}_G^{\text{total}} = \mathcal{L}_G + \mathcal{L}_{L_1} + \mathcal{L}_{L_2} \tag{5}$$

3.2. CLIP

A unique challenge in our project is developing or adapting a text embedding strategy that captures the emotive and thematic content of lyrics in a way that is useful for conditioning image generation. To address this, we incorporate CLIP-based augmentations in two ways: PromptFusion embeddings and a contrastive CLIP loss.

PromptFusion Embedding (CLIP-based). This method uses the CLIP text encoder (ViT-B/32) to produce a structured 1536-dimensional embedding by encoding three distinct prompts:

- "Album title: *X*"
- "Lyrics summary: *Y*" (generated by LongT5)
- "Top three sentiments: joy, sadness, anger" (predicted by RoBERTa)

Each prompt is tokenized and encoded independently. The resulting 3 vectors (each 512-D) are concatenated:

$$\varphi(t) = \text{CLIP}(t_{\text{title}}) \parallel \text{CLIP}(t_{\text{summary}}) \parallel \text{CLIP}(t_{\text{sentiments}})$$

This design preserves semantic separation and leverages CLIP's joint vision-language space without fine-tuning.

CLIP-Based Contrastive Loss. In models using CLIP embeddings, we optionally add a contrastive loss that aligns the generated image and input prompt in CLIP's joint vision–language space to the total generator loss given by Equation 5:

$$\mathcal{L}_{\text{CLIP}} = -\lambda_{\text{clip}} \cdot \cos\left(\psi(G(z, \varphi(t))), \varphi_{\text{CLIP}}(t)\right) \tag{6}$$

Here, $\psi(\cdot)$ is the frozen CLIP image encoder, and $\varphi_{\text{CLIP}}(t)$ is the normalized CLIP-encoded text prompt.

3.3. Refined GAN

The refined GAN builds upon the existing baseline model with several enhancements for training stability and sample fidelity. Table1 highlights the key design choices and differences.

Spectrally Normalized GAN with CLIP Conditioning and Augmentation. The generator receives a noise vector z and a 1536-dimensional CLIP embedding $\varphi(t)$. Embedding is projected via a single-layer MLP and concatenated with z. A dense layer maps the joint vector to a $4 \times 4 \times 512$ tensor. Upsampling is performed via bilinear upsampling followed by 3×3 convolutions, batch normalization, and ReLU. This avoids checkerboard artifacts associated with transposed convolutions. The number of upsampling blocks is adaptive to image size, supporting 64×64, 128×128, and 256×256 outputs.

The descriminator processes inputs through a stack of spectral-normalized convolutional blocks with progressive downsampling to 4×4. Global average pooling is applied to the final feature map to extract a 512-D image representation. The CLIP embedding is projected into the same space and combined with the pooled feature via an inner product, forming a conditional score. A second real/fake score is computed directly from the feature map via a 4×4 spectral-normalized convolution, and the final discriminator output is the sum of both.

AugmentPipe and ADA. We incorporate an adaptive augmentation pipeline that applies color jitter, rotation, and blur with a tunable augmentation strength $p \in [0, 1]$. The value of p is dynamically adjusted during training based on the discriminator's confidence on real data, following Karras et al. [6]. The augmentation is differentiable and mixes original and transformed images:

$$x_{\text{aug}} = (1 - p) \cdot x + p \cdot \mathcal{T}(x)$$

Mixed Precision and Checkpointing. The refined GAN supports automatic mixed precision (AMP) and includes robust checkpointing of model weights, optimizer states, and ADA parameters, enabling safe recovery and reproducibility.

3.4. Quantitative Metrics

Since adversarial training losses (e.g., generator and discriminator loss) are known to correlate poorly with perceptual quality and reflect progress of the models, we rely on two widely used quantitative metrics to assess the fidelity and diversity of the generated album covers: the *Inception Score* (IS) and the *Fréchet Inception Distance* (FID). IS evaluates how confidently and distinctly images are classified by an Inception network, thereby capturing both image quality and diversity, while FID measures the distributional similarity between generated and real images in a pretrained feature space.

Table 1. Comparison of the Baseline cDCGAN and our refined variant.

Component	Baseline	Refined
Text Embedding	MiniLM (3 × 384-D)	CLIP ViT-B/32 (3 × 512-D)
Embedding Fusion	Concatenation	Concatenation
Generator Upsampling	Transposed conv	Bilinear upsampling + conv
Discriminator Downsampling	Strided conv + BN	Spectral norm conv
Condition Injection	Spatial concat @ 4×4	Inner product after pooling
Spectral Normalization	x	✓
Adaptive Data Aug. (ADA)	x	✓
Mixed Precision (AMP)	x	✓
Loss Function	BCE + L_1 + L_2	BCEWithLogits + L_1 + L_2

Inception Score (IS). IS [12] evaluates both the visual quality and diversity of generated samples. It is defined as:

$$\text{IS} = \exp\left(\mathbb{E}_{\mathbf{x} \sim p_g}\left[D_{\text{KL}}(p(y|\mathbf{x}) \| p(y))\right]\right), \tag{7}$$

where $p(y|\mathbf{x})$ is the predicted label distribution from an Inception classifier for a generated image $\mathbf{x}$, and $p(y)$ is the marginal class distribution. High IS indicates that generated images are both sharp (low-entropy $p(y|\mathbf{x})$) and diverse (high-entropy $p(y)$).

Fréchet Inception Distance (FID). FID [4] measures the similarity between the distributions of real and generated images in the feature space of a pretrained Inception network. Assuming both feature distributions are Gaussian, FID is computed as:

$$\text{FID} = \|\mu_r - \mu_g\|^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}\right), \tag{8}$$

where (μ_r, Σ_r) and (μ_g, Σ_g) denote the means and covariances of the real and generated image features, respectively. Lower FID indicates better alignment with real image statistics.

4. Experiments

We evaluate five model variants that systematically vary in architecture, text conditioning method, and the inclusion of CLIP-based contrastive loss. These include: the baseline conditional GAN with MiniLM embeddings, a version with CLIP embeddings, and a further extension with CLIP contrastive loss; as well as our proposed refined GAN architecture with and without contrastive loss (see table 2 for summary).

All models are trained for 350 epochs with a batch size of 64 in consideration of training speed and cost. We use the Adam optimizer ($\beta_1 = 0.5$, $\beta_2 = 0.999$) with ($\beta_1 = 0.5, \beta_2 = 0.999$), a common choice in GAN literature for stabilizing adversarial updates. For the refined models, we adopt the Two-Time-Scale Update Rule (TTUR), using separate learning rates for the generator (1×10^{-4}) and

discriminator (4×10^{-4}) to encourage faster convergence of the discriminator in early training. We use mixed precision (AMP) to improve training speed and memory efficiency. The reconstruction loss coefficients are fixed at $\lambda_{L_1} = 50$ and $\lambda_{L_2} = 100$ to strongly regularize the generator toward structural realism and feature-level fidelity. The CLIP contrastive loss coefficient is annealed from 5.0 to 15.0 across training epochs to allow early-stage convergence before introducing stronger semantic alignment signals.

Table 2. Summary of model variants evaluated. We compare two architectures (Baseline vs. Refined), two types of text embedding (MiniLM vs. CLIP), and optional CLIP-based contrastive loss.

Model Name	Arch.	Embedding	CLIP Loss
Baseline	Baseline	MiniLM	x
Baseline + CLIP	Baseline	CLIP	x
Baseline + CLIP Loss	Baseline	CLIP	✓
Refined	Refined	CLIP	x
Refined + CLIP Loss	Refined	CLIP	✓

We report FID and IS computed on the same held-out validation set across all five model variants every 50 epochs for 350 epochs to assess how different architectural and conditioning choices affect generation performance. Note that higher IS and lower FID are preferred.

Inception Score (IS). As shown in Figure 1, all models exhibit steady improvement in Inception Score during the early stages of training, though notable differences emerge as training progresses. The **Refined + CLIP Loss** model consistently achieves the highest IS, reaching above 4.8 by epoch 350, which suggests it generates higher-quality and more semantically diverse images. The **Refined** model without CLIP loss also performs competitively, highlighting the effectiveness of architectural enhancements such as spectral normalization, bilinear upsampling, and adaptive data augmentation.

Within the baseline family of models, switching from MiniLM to CLIP embeddings alone does not seem to alter the performance much. However, incorporating the CLIP-based contrastive loss yields further gains, even without architectural changes—demonstrating that the loss component adds measurable value. Compared to the refined models, baseline variants plateau earlier and at lower IS scores, indicating their more limited ability to generate diverse and coherent visual content.

Fréchet Inception Distance (FID). As shown in Figure 2, the FID curves support the trends observed in IS while also highlighting important distinctions. The **Refined + CLIP Loss** model consistently achieves the lowest FID, converging to around 135 by epoch 300, indicating the strongest alignment with the real image distribution. The

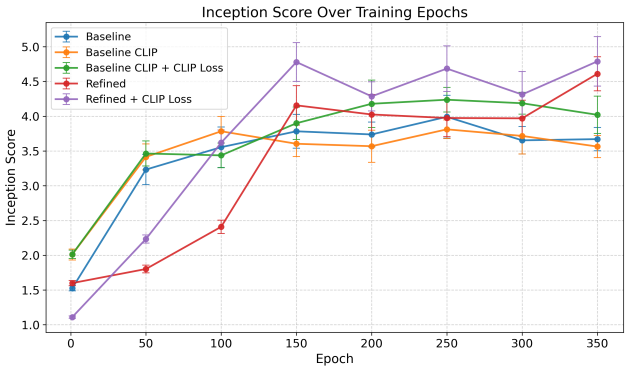

Figure 1. Inception Score (IS) over training epochs as well as the standard deviations for all model variants. Higher values indicate better image quality and diversity.

Refined model also performs competitively, reaching similar values near the end of training and outperforming all baseline variants.

Among the baseline models, the **Baseline + CLIP** variant shows slightly better performance than the one augmented with contrastive loss. While **Baseline + CLIP + CLIP Loss** initially improves over the vanilla baseline, its FID degrades slightly in later epochs, suggesting potential training instability introduced by the additional loss component. The **Baseline** model, which featuring a sharper decrease of FID in the beginning, exhibits increasing FID since epoch 100, indicating a more limited capacity to approximate the real image distribution compared to the refined models.

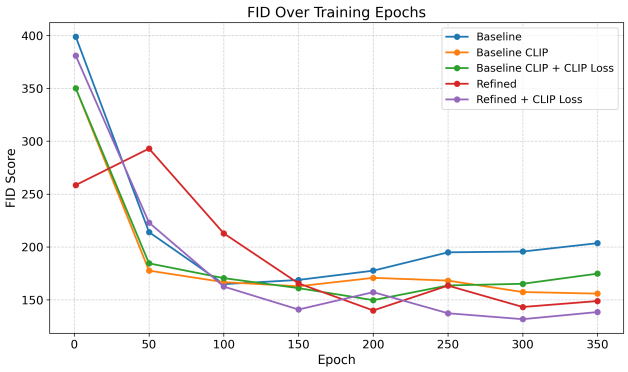

Figure 2. Fréchet Inception Distance (FID) over training epochs for all model variants. Lower values indicate that generated images are closer to real images in distribution.

These results demonstrate that both CLIP-based conditioning and architectural improvements contribute significantly to generation quality. The best performance is achieved by combining both in the **Refined + CLIP Loss** model, which shows robust improvements across both IS and FID (Figure 3 and 4).

In addition to quantitative evaluation, we present quali-

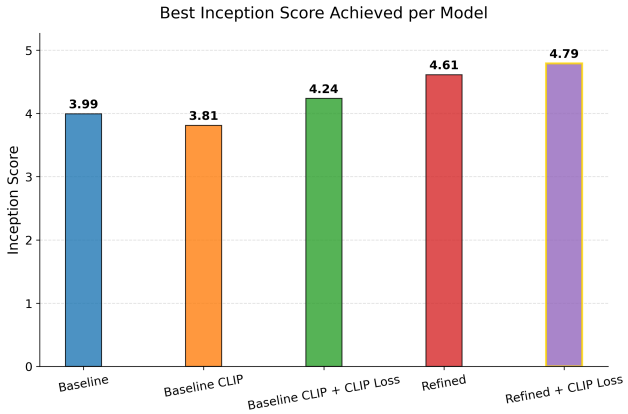

Figure 3. Best Inception Score (IS) achieved by each model over all 350 training epochs, with the bar outline of the best-performing model highlighted in yellow.

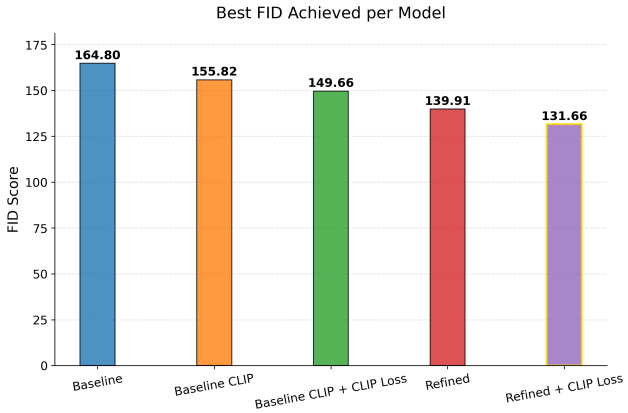

Figure 4. Best Fréchet Inception Distance (FID) achieved by each model over all 350 training epochs, with the bar outline of the best-performing model highlighted in yellow.

tative results from each model variant by visualizing samples from the best-scoring epoch checkpoint, allowing for a direct visual comparison of generative capabilities across conditions.

Figures 6 through 10 present album covers generated by the five model variants for the same set of 16 randomly chosen albums with their corresponding real cover images shown in 5. The **baseline model** (Figure 6) often produces noisy, abstract textures with minimal structure. Many images exhibit washed-out or dull color palettes and lack meaningful compositional coherence, and a few of the generated images are not very distinguishable from one another, indicating limited diversity.

The **Baseline + CLIP** variant (Figure 7) shows slight improvement in contrast and texture definition. Some images exhibit more saturated colors, such as blues and reds, though object structure remains indistinct and some artifacts persist.

Figure 5. A Sample of 16 Real Album Covers

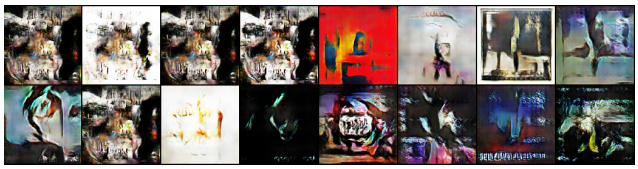

Figure 6. Covers Generated by Baseline model

Figure 7. Covers Generated by Baseline + CLIP Embeddings model

Figure 8. Covers Generated by Baseline + CLIP Embeddings + CLIP Loss model

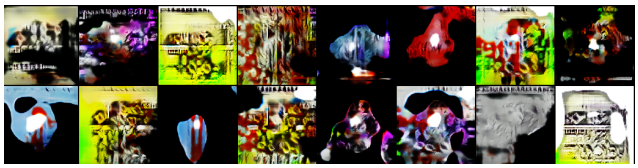

Figure 9. Covers Generated by Refined model

Figure 10. Covers Generated by Refined + CLIP Loss model

Introducing the **CLIP contrastive loss** (Figure 8) leads to moderate gains in sharpness and local structure. While the generated images are still abstract, a few samples suggest loosely interpretable forms—such as symmetrical patches, central focal blobs, or faint outlines. Color usage also appears more varied.

The **Refined model without CLIP loss** (Figure 9) produces cleaner textures and more consistent visual patterns.

Bilinear upsampling and architectural modifications appear to reduce checkerboard artifacts, and a few images display bold shapes and centered color blobs that resemble abstract cover design motifs.

Finally, the **Refined + CLIP model** (Figure 10) offers the most coherent samples overall. Colors are more vivid, several images show distinct zones or layered patterns, and some exhibit loosely frame-like or portrait-like symmetry. Despite these improvements, most images remain abstract and do not yet reach photorealism or detailed semantic alignment.

Overall, we observe that architectural refinements and CLIP-based supervision contribute complementary improvements—particularly in terms of visual coherence, saturation, and abstract structure.

5. Conclusions

In this work, we systematically evaluated how architectural design choices, text embedding strategies, and multimodal supervision affect the quality of text-conditioned image generation for the task of album cover synthesis. Starting from a baseline conditional GAN using MiniLM embeddings, we introduced CLIP-based conditioning and a refined GAN architecture that incorporates spectral normalization, adaptive data augmentation, and improved conditioning injection.

Our experiments demonstrate that each enhancement contributes to improved generative performance, as measured by both Fréchet Inception Distance (FID) and Inception Score (IS). The refined architecture proves more robust and stable across training, while CLIP-based embeddings, when paired with a contrastive loss, significantly improve semantic alignment between text and image. Notably, the **Refined + CLIP Loss** model consistently achieves the best quantitative results and produces more coherent, visually distinct outputs. These findings highlight the importance of combining principled architectural refinements with semantically rich conditioning for text-to-image synthesis tasks.

While our CLIP-guided GAN framework demonstrates interesting results, several directions could further improve performance and flexibility. Future work may explore more effective fusion of textual modalities using cross-attention or multi-branch conditioning to better capture the roles of lyrics, title, and sentiment. Hierarchical or multi-stage architectures could improve compositional coherence and visual detail. Beyond generation, supporting user-guided editing through prompts or interactive feedback—potentially via diffusion models or reinforcement learning—would enable greater control. More comprehensive evaluation, including more human reviews of generated results or task-specific metrics, could offer deeper insights. Finally, expanding the dataset to include more albums, genres, languages, or style-specific models may enhance diversity and generalizability.

References

[1] E. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep generative image models using a laplacian pyramid of adversarial networks, 2015.

[2] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks, 2014.

[3] A. Hepburn, R. McConville, and R. Santos-Rodríguez. Album cover generation from genre tags. In *Proceedings of the 10th International Workshop on Machine Learning and Music*, Barcelona, Spain, 2017.

[4] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018.

[5] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation, 2018.

[6] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila. Training generative adversarial networks with limited data. In *Advances in Neural Information Processing Systems*, volume 33, pages 12104–12114. Curran Associates, Inc., 2020.

[7] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks, 2019.

[8] J. Marien, S. Leroux, B. Dhoedt, and C. D. Boom. Audio-guided album cover art generation with genetic algorithms, 2022.

[9] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021.

[10] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

[11] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis, 2016.

[12] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans, 2016.

[13] F. P. Stoppa, E. Vidaña-Vila, and J. Navarro. Album cover art image generation with generative adversarial networks, 2022.