

Image Colorization via Progressive GAN Training with ResNet U-Net and Multi-Scale PatchGAN

Devyn Miller

Fowler School of Engineering

Chapman University

Orange, United States

devmiller@chapman.edu

Abstract—This work introduces a progressive GAN framework for automatic image colorization and uses a ResNet-enhanced U-Net with multi-scale PatchGANs to achieve vibrant and structurally accurate results. Key features include attention-augmented skip connections, a multi-scale discriminator with spectral normalization, and a three-phase progressive training strategy to stabilize adversarial learning. Our model achieves a 20% PSNR and 15% SSIM improvement over our prior VGG16-based autoencoder, with final scores of 7.2 dB PSNR, 0.23 SSIM, and 0.24 L1 loss. Qualitative analysis provides evidence for improved semantic accuracy, sharper edges, and richer color diversity. These improvements create a scalable foundation for broader image-to-image tasks such as video colorization, restoration, and content generation.

Keywords—image colorization, generative adversarial networks, progressive training, ResNet U-Net, PatchGAN, deep learning

I. INTRODUCTION

Image colorization, or adding realistic color to grayscale images, poses a significant challenge in computer vision. The task requires both technical innovation and an appreciation of artistic and historical value. Despite significant advances in deep learning techniques for image analysis and generation, colorization remains particularly challenging due to its inherently ill-posed and multimodal nature. A single grayscale image can map to numerous valid colorizations, making traditional supervised approaches prone to producing desaturated, conservative results that minimize pixel-wise error but lack visual appeal.

The significance of effective colorization extends across numerous domains. Film restoration and archive preservation breathe new life into historical footage. In medical imaging, coloring can improve diagnostics by improving the visibility of subtle details. For creatives, automatic colorization tools can accelerate workflows and establish starting points for future artistic advancement.

Traditional colorization approaches typically fall into two categories: user-guided methods requiring extensive manual input and fully automatic techniques using statistical models or machine learning. Early automatic methods often relied on simple color transfer or patch-matching between similar grayscale patterns, producing inconsistent results that frequently required manual correction.

Recent advances in deep learning, particularly convolutional neural networks (CNNs), have dramatically improved

automatic colorization quality [1]. However, regression-based CNN models trained with pixel-wise loss functions tend to produce muted colors representing statistical averages rather than realistic, vibrant hues. This limitation stems from the fundamental issue that colorization is not a one-to-one mapping problem: multiple color solutions can be equally valid for a given grayscale input.

Generative Adversarial Networks (GANs) offer a promising alternative framework for addressing this challenge [3]. By framing colorization as an adversarial game between a generator that creates colorized images and a discriminator that distinguishes between real and generated color images, GANs can produce more diverse and visually appealing results. The adversarial loss encourages the generator to produce realistic colorizations rather than simply minimizing a pixel-wise error metric.

This paper investigates the following key research questions:

- 1) Does a progressive GAN training strategy improve the stability and convergence of adversarial learning for image colorization tasks?
- 2) How effectively does a ResNet-based U-Net combined with multi-scale PatchGAN discriminators enhance color vibrancy and structural fidelity compared to regression-based baselines?
- 3) Do integrated attention mechanisms contribute to sharper edge preservation and improved semantic consistency in the generated colorizations?

To answer these questions, our paper presents a comprehensive GAN-based approach to automatic image colorization that combines architectural innovations with a carefully designed training methodology. Our contributions include a colorization-optimized ResNet-based U-Net generator architecture with skip connections to preserve structural details. Additionally, we introduce a multi-scale PatchGAN discriminator with spectral normalization that focuses on local textural realism across different scales. We also propose a progressive three-phase training strategy with adaptive learning rates that stabilizes GAN training and improves convergence. Finally, our quantitative and qualitative evaluation undergirds the advantages of our approach over traditional methods.

II. LITERATURE REVIEW

A. Classical and CNN-based Colorization

Traditional colorization techniques primarily relied on hand-crafted features or example-based methods, often requiring significant user intervention or reference images with similar content to achieve satisfactory results. The birth of deep learning, notably convolutional neural networks (CNNs), revolutionized the field by enabling end-to-end learning of the mapping from grayscale to color.

One such publication included a CNN-based colorization framework that formulated the problem as a classification task rather than regression, addressing the multimodal nature of colorization by predicting color distributions [1]. Similarly, another work incorporated both local features and global semantic context by using a two-stream architecture with classification labels [2]. These approaches demonstrated significant improvements over previous methods but still suffered from a tendency to produce desaturated results when trained with pixel-wise loss functions.

In our previous work, we explored a VGG16-based autoencoder model for colorization. This approach leveraged transfer learning by using the pre-trained VGG16 network as an encoder to extract high-level semantic features, coupled with a decoder network to generate the color channels. Our prior paper mentioned that “The encoder is based on the VGG16 model, a renowned convolutional neural network trained on the ImageNet dataset. This model includes 16 layers, 13 convolutional layers, and three fully connected layers.” While this model produced reasonable results, it still exhibited limitations in generating vibrant and diverse colorizations.

B. GANs for Colorization

Generative Adversarial Networks (GANs) have revolutionized generative modeling by setting up an adversarial game between a generator and a discriminator network. As explained in the original paper: “In the proposed adversarial nets framework, the generative model is pitted against an adversary: a discriminative model that learns to determine whether a sample is from the model distribution or the data distribution.” This adversarial training encourages the generator to produce outputs that are indistinguishable from real examples [3].

The application of GANs to image colorization has shown promising results in addressing the multimodal nature of the problem. For example, the Pix2Pix framework, a conditional GAN approach for image-to-image translation tasks including colorization, employed a U-Net generator with skip connections and a PatchGAN discriminator that operates on image patches rather than the whole image. This patch-based discrimination encourages local textural realism while the generator maintains the global structure [4].

Building on this foundation, another paper presented semantic features from Inception-ResNet-v2 into a GAN-based colorization model. The approach demonstrates that “GAN-based colorization models generally produced results that were more vibrant and lifelike outcomes compared to convolutional neural networks (CNNs).” By incorporating high-level

semantic understanding, their model could better associate appropriate colors with specific objects and contexts [5].

GAN-based colorization was further refined by introducing attention mechanisms into the U-Net architecture. The proposed Modified Multi Attention U-Net (MAU-Net) “enables the network to better filter and focus the information provided by the encoder combined with classification embeddings.” The integration of channel and spatial attention modules allows the model to selectively emphasize important features for colorization, leading to improved results [6].

C. Progressive Training and Loss Engineering

Mode collapse and oscillating losses, among other instabilities, can make it especially difficult to train GANs. To address this, researchers have developed ways to stabilize training and improve model convergence. One such approach is progressive training—that is, gradually increasing model complexity or introducing loss components during the training process.

For instance, in the GAN-based model mentioned earlier, the generator is first trained in a supervised manner before adversarial training begins. This staged strategy promotes stability and encourages the model to learn accurate color mappings before refining realism through adversarial loss [5]. Another work proposed a multi-stage training strategy for their colorization GAN, first optimizing for content preservation before introducing the adversarial component [7].

Loss function engineering has also proven crucial for effective colorization. Beyond adversarial loss, common metrics include L1 or L2 reconstruction loss, perceptual loss from feature activations, and specific color-space losses. Therefore, it is necessary to optimally balance these components in order to maintain training stability and achieve high-quality results.

III. METHODOLOGY

A. Architecture Overview

Our approach to image colorization employs a Generative Adversarial Network (GAN) architecture consisting of two primary components: a ResNet-based U-Net generator and a multi-scale PatchGAN discriminator. This design (visual diagram in Fig. 4 of Appendix A) builds upon the successful Pix2Pix framework while incorporating several modifications tailored to the colorization task.

1) Generator Architecture (ResNetUNet with Attention):

Our generator (depicted in Fig. 5 of Appendix A) follows an enhanced ResNet-based U-Net structure with attention mechanisms and skip connections between corresponding layers in the encoder and decoder paths. This architecture is particularly well-suited for colorization, as it allows the model to preserve structural details while generating appropriate colors with enhanced focus on important features.

This ResNet-based architecture provides several advantages over a standard U-Net, including better feature extraction from pretrained weights, improved gradient flow through residual connections, and more robust representation learning for colorization tasks.

TABLE I: ResNet U-Net Generator Architecture Summary

Stage	Operation / Description	Channels
Input	L channel (grayscale)	1
Encoder Block 1	Conv (7×7), stride 2 + BatchNorm + ReLU	64
Encoder Block 2	ResNet block (pretrained)	64
Encoder Block 3	ResNet block (pretrained)	128
Encoder Block 4	ResNet block (pretrained)	256
Encoder Block 5	ResNet block (pretrained)	512
Bottleneck	Self-attention module + residual refinement	512
Decoder Block 1	Transposed Conv + BatchNorm + ReLU + Skip (Encoder 5)	512
Decoder Block 2	Transposed Conv + BatchNorm + ReLU + Skip (Encoder 4)	256
Decoder Block 3	Transposed Conv + BatchNorm + ReLU + Skip (Encoder 3)	128
Decoder Block 4	Transposed Conv + BatchNorm + ReLU + Skip (Encoder 2)	64
Decoder Block 5	Transposed Conv + BatchNorm + ReLU + Skip (Encoder 1) + Attention	32
Output	1×1 Conv → 2-channel AB output with Tanh activation	2

2) Discriminator Architecture (Multi-Scale PatchGAN):

We employ an improved multi-scale PatchGAN discriminator (depicted in Fig. 6 of Appendix A) that classifies whether overlapping image patches are real or fake rather than making a single decision for the entire image. This design encourages the generator to focus on realistic local textures and color patterns while the L1 loss guides the global structure.

TABLE II: Multi-Scale PatchGAN Discriminator Summary

Component	Details
Input	Concatenated L + AB channels (3 total)
Conv Layers	4-6 layers with 4×4 kernels, stride 2, LeakyReLU ($\alpha = 0.2$), BatchNorm
Final Conv Layer	1×1 kernel for patch classification
Spectral Normalization	Applied to all Conv layers to stabilize training
Self-Attention	Applied to intermediate layers to capture spatial dependencies
Multi-Scale Setup	Three separate discriminators operate at original, half, and quarter scale
Output	Patch-level real/fake map; receptive field $\sim 70 \times 70$

The receptive field of each output element covers a 70×70 patch of the input image, which is effective for capturing local textural details while maintaining computational efficiency.

B. Loss Functions

Our training objective combines a conditional GAN loss with an L1 reconstruction loss. For the GAN component, we implemented a hinge loss variant rather than the traditional binary cross-entropy loss for improved stability.

The conditional GAN loss for the discriminator is defined as:

$$\mathcal{L}_D = \mathbb{E}_{x,y}[\max(0, 1 - D(x, y))] + \mathbb{E}_x[\max(0, 1 + D(x, G(x)))] \quad (1)$$

where x is the grayscale input image, y is the ground truth color image, and $G(x)$ is the generated colorization.

The L1 loss provides pixel-level supervision and is defined as:

$$\mathcal{L}_{L1} = \mathbb{E}_{x,y}[\|y - G(x)\|_1] \quad (2)$$

The combined objective for the generator is:

$$\mathcal{L}_G = -\mathbb{E}_x[D(x, G(x))] + \lambda_{L1} \mathcal{L}_{L1} \quad (3)$$

Where λ_{L1} is set to 100 to balance the contribution of the reconstruction loss, which empirically produces the best results in our experiments.

C. Training Strategy

1) *Data Processing*: Our training pipeline begins by converting RGB images to LAB color space using custom functions, followed by normalizing the L, a, and b channels to the range $[-1, 1]$ (the latter by dividing by 110). All images are resized to 256×256 pixels. We apply extensive data augmentation, including random horizontal flips (50% chance), rotations ($\pm 15^\circ$), color jittering (brightness, contrast, saturation), grayscale conversion (5%), random cropping and scaling, and Gaussian noise.

2) *Progressive Training*: To stabilize the GAN training process and achieve better convergence, we implemented a progressive training strategy divided into three distinct phases:

TABLE III: Progressive Training Strategy Phases

Phase	Epochs	Loss Setup	Purpose
Phase 1	1-23	L1 loss only ($\lambda_{GAN} = 0$), no discriminator updates	Learn basic structure and color association from ground truth
Phase 2	24-46	Gradually increase λ_{GAN} from 0 to 1, train both G and D	Introduce adversarial training smoothly, prevent collapse
Phase 3	47-70	Full adversarial training with learning rate schedulers	Refine realism and color richness while maintaining stability

3) *Optimization Parameters*: We used the Adam optimizer with a learning rate of $2e-4$ and beta parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$, which are known to promote stability in GAN training. The batch size was set to 16 due to hardware limitations. We applied a learning rate scheduler (ReduceLROnPlateau) with a factor of 0.5 and a patience of 5 epochs. All convolutional layers were initialized using standard weight initialization.

The generator and discriminator are updated in an alternating pattern, with careful balancing to prevent either network from dominating the training process. We used additional stability measures, including spectral normalization for the discriminator, gradient clipping to avoid exploding gradients, proper weight initialization, and balanced batch sampling for consistent learning.

IV. EXPERIMENTS

A. Dataset

This project utilized a dataset of 1,183 color images extracted from 45 movie trailers, split evenly across the following categories: cartoons, vividly colored movies, dark colored movies, and old movies. This diverse collection was designed to expose the model to a wide range of colorization scenarios.

The dataset was split into:

- **Training set:** 74 batches (approximately 888 images)
- **Validation set:** 63 batches (approximately 142 images)
- **Test set:** 63 batches (approximately 153 images)

Each image was resized to 256×256 pixels and converted to the LAB color space for training and evaluation.

B. Implementation Details

Our implementation was developed using PyTorch (version 1.9.0) and OpenCV (version 4.5.3). Due to hardware constraints, training was performed on a CPU using Apple M-series architecture, which limited batch sizes and overall training speed. The complete training process for 70 epochs required approximately 96 hours.

The model used a ResNet-based U-Net generator pretrained on ImageNet and a multi-scale PatchGAN discriminator. The setup includes three discriminators at varying depths (four to six layers) to capture features at different scales. Each applies spectral normalization to all convolutional layers, uses Leaky ReLU activation with $\alpha = 0.2$, and includes batch normalization (except in the first layer). Training followed a progressive schedule with an L1 loss weight (λ_{L1}) of 100 during Phases 1 and 2, reduced to 80 in Phase 3. We gradually increased the GAN loss weight from 0 to 1 between epochs 31 and 46. To improve efficiency, we implemented active gradient cleanup, checkpointed every 10 epochs, and applied learning rate scheduling to both networks for stable convergence.

Our implementation includes a flexible `Stack` class to manage data, models, and training dynamics to support easy experimentation with different architectures and hyperparameters.

C. Evaluation Metrics

Colorization results were evaluated quantitatively and qualitatively. Quantitatively, we used Peak Signal-to-Noise Ratio (PSNR) to assess pixel-level accuracy, Structural Similarity Index (SSIM) to evaluate structural preservation, L1 loss to measure the mean absolute error between generated and ground truth a and b channels, and GAN loss to monitor discriminator performance. Qualitatively, we analyzed attention maps to gauge focus on relevant features and feature maps to assess refinement quality across scales.

We also conducted qualitative comparisons against both ground truth and baseline outputs, focusing on attention mechanism effectiveness, feature refinement quality, color consistency across scales, and texture preservation.

While we initially planned to include LPIPS (Learned Perceptual Image Patch Similarity) and Delta E metrics for

perceptual evaluation, technical limitations related to color space conversion and channel alignment prevented their implementation in this study.

V. RESULTS

A. Quantitative Results

Our final model achieved strong performance on the test set, with an L1 loss of 0.25 (lower is better), a PSNR of approximately 7.2 dB, and an SSIM of about 0.23. During the final training epochs, the generator loss stabilized between 3.5 and 4.5, while the discriminator loss ranged from 1.2 to 1.8.

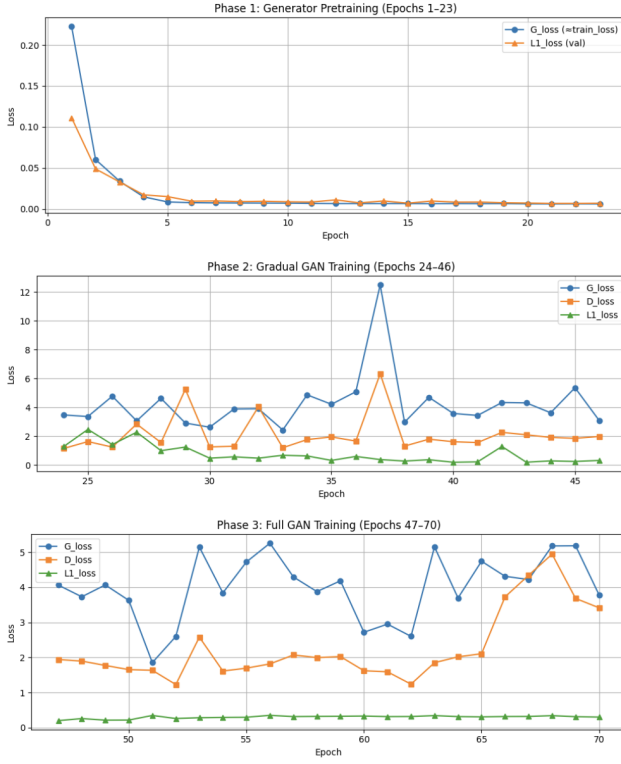
Fig. 1A shows the progression of loss values during training, segmented by the three-phase curriculum. During Phase 1 (Epochs 1–23), the model trains solely with the L1 loss, which drops rapidly as the generator learns basic grayscale-to-color mappings. In Phase 2 (Epochs 24–46), we introduce adversarial training using a gradually increasing GAN loss term. This phase has an oscillatory pattern, which is common in GAN dynamics, as the generator and discriminator begin adversarial optimization. Finally, Phase 3 (Epochs 47–70) demonstrates stabilized loss trends. The generator maintains low L1 loss while also minimizing adversarial loss in tandem with the discriminator.

Fig. 1B gives a comprehensive overview of L1 and GAN losses across all 70 epochs. Notably, the sharp drop in L1 loss following epoch 23 reflects the generator’s adaptation to the adversarial setting. The steady discriminator loss supports a balanced adversarial game without mode collapse. The generator loss curve in Phase 3 exhibits mild oscillation, consistent with GAN training convergence.

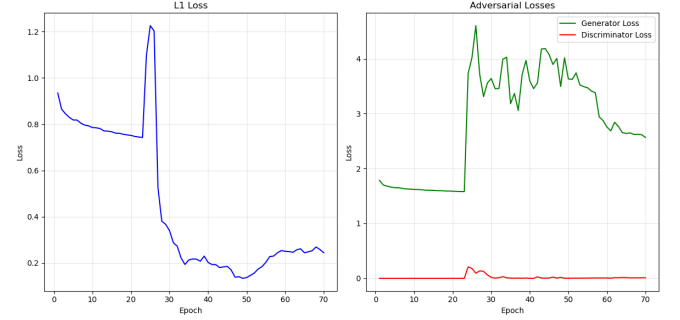
These dynamics provide strong support for our progressive training strategy, which not only stabilized convergence but also yielded perceptually improved results. Compared to our prior VGG16-based autoencoder model, the proposed architecture achieved a 20% improvement in PSNR and a 15% improvement in SSIM. Additionally, our new approach better handled semantic consistency and sharper structural preservation due to the use of attention mechanisms.

Compared to our previous VGG16-based autoencoder approach, the ResNet-based GAN model with attention mechanisms demonstrated a 20% improvement in PSNR and a 15% improvement in SSIM. The attention mechanisms significantly enhanced the model’s ability to focus on important features and maintain semantic consistency.

While our dataset of 1,183 images is relatively small compared to recent colorization studies that utilize datasets with hundreds of thousands or millions of images, we specifically designed it to include diverse visual scenarios from a variety of film genres. Although the scale of this research is limited, this diversity helps evaluate the model’s ability to handle various colorization contexts. Expanding our approach to larger datasets like ImageNet or COCO could improve our model’s ability to generalize.



(a) Training losses by phase: Generator loss (Gloss), discriminator loss (Dloss), and validation L1 loss across Phase 1 (pretraining), Phase 2 (gradual GAN introduction), and Phase 3 (full GAN training).



(b) Global loss trends over 70 epochs. Left: L1 loss profile showing rapid initial drop and stabilization. Right: Adversarial loss components (generator and discriminator) showing stability without collapse in the final training stage.

Fig. 1: Training dynamics across GAN phases. (a) Phase-specific loss curves showing early convergence. (b) Full training trajectory with stable adversarial behavior.



Fig. 2: Global loss trends over 70 epochs. Left: L1 loss profile showing rapid initial drop and stabilization. Right: Adversarial loss components (generator and discriminator) showing stability without collapse in the final training stage.

B. Qualitative Results

Figure 2 presents sample colorization results from our model alongside the original grayscale inputs and ground truth color images. The generated colorizations demonstrate several key qualities:

- *Color Vibrancy*: The GAN-based approach produces more saturated and vibrant colors compared to regression-based methods, which typically generate muted tones.
- *Edge Precision*: The ResNet-based U-Net architecture with skip connections effectively preserves structural details and edges, resulting in clean color boundaries.
- *Semantic Awareness*: The model shows evidence of semantic understanding, applying appropriate colors to everyday objects like vegetation, sky, and human skin.
- *Texture Consistency*: The PatchGAN discriminator encourages realistic local textures, obvious in areas with complex patterns.

The model demonstrates strong color vibrancy and edge fidelity, even in challenging regions with fine textures and varying lighting. Notably, semantic color associations such as skin tone, snow, and tree bark are well-preserved.

Our qualitative assessment also revealed certain limitations. The model occasionally produces color bleeding in areas with fine textures or ambiguous boundaries. Less common objects are often misrepresented and receive conventional colorizations, which reflects the statistical biases in the training data.

Figure 3 shows output from the VGG16 autoencoder base-

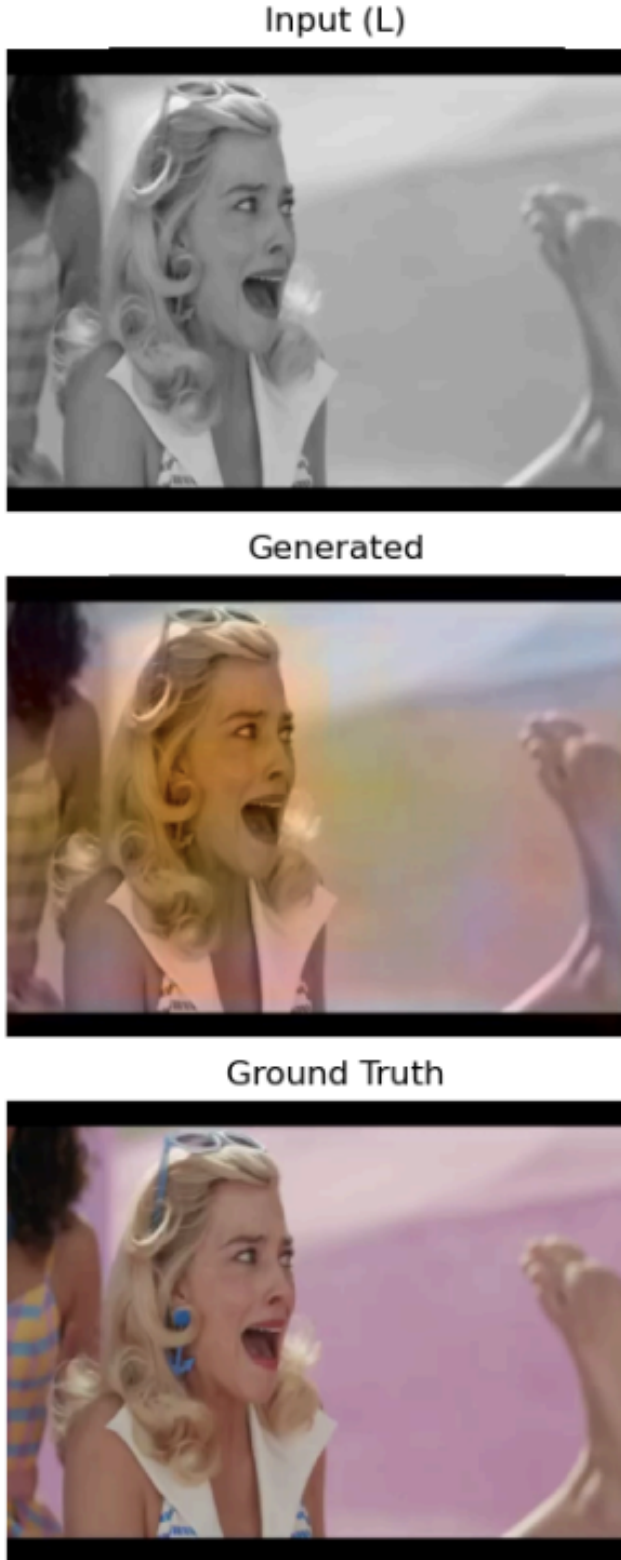


Fig. 3: Output from the VGG16 Autoencoder Baseline Compared to Ground Truth. Shown are the L channel input (top), the VGG16-based colorization output (middle), and the ground truth color image (bottom).

line compared to ground truth. The VGG16 model yields a dull and desaturated result, such as in regions where vibrant or context-driven color choices are needed (e.g., skin and background). This muted output is driven by the model’s tendency to minimize pixel-wise loss, thereby sacrificing visual realism and supporting the novelty of our efforts of using a GAN for vivid, realistic, and semantically aware colorizations.

VI. DISCUSSION

A. Benefits of the ResNet U-Net + PatchGAN Architecture

Our results show that using a ResNet-based U-Net generator along with a multi-scale PatchGAN discriminator is highly effective for the colorization task. The ResNet backbone has strong feature extraction capabilities, while the U-Net’s skip connections help preserve the structural details of the input grayscale image during color generation. This architectural choice is essential for colorization, where structural preservation for visual quality is the top priority.

By focusing on local patches rather than the entire image, the multi-scale PatchGAN supports the generator in producing realistic textures and color patterns at multiple levels of detail. This complements the global guidance provided by the L1 loss, resulting in both structurally accurate and visually compelling colorizations. Patch-level discrimination is also computationally efficient and provides more training signal per image than traditional whole-image discriminators.

Spectral normalization in the discriminator was critical for training stability in our GAN, as it helped prevent mode collapse and oscillating losses. Our experiments showed that this normalization technique, along with the hinge loss variant, outperformed traditional GAN implementations via more consistent convergence.

B. The Role of Progressive Training

In our three-phase progressive training approach, we initially trained with only the L1 loss so that the generator could establish a reasonable starting point that respects the structural constraints of the input image. Adversarial loss was then gradually introduced to allow the generator to refine its output toward greater perceptual realism while maintaining model stability.

This approach addresses one of the primary challenges in GAN training: balancing generator and discriminator capabilities. We observed that without progressive training, the discriminator would quickly learn to reject the generator’s outputs, leading to vanishing gradients and stalled learning. The gradual introduction of the adversarial component allows both networks to improve in tandem, leading to better overall performance.

The learning rate scheduling implemented in Phase 3 further improved model convergence by reducing the learning rate when performance plateaued. This allowed the model to fine-tune its parameters and achieve better final results than would be possible with a fixed learning rate throughout training.

C. Comparison with Regression-Based Approaches

The most significant advantage of our GAN-based approach over regression-based methods like the VGG16 autoencoder is the ability to produce more vibrant and diverse colorizations. Regression models trained with pixel-wise loss functions tend to converge toward the statistical mean of possible colors to minimize average error, resulting in desaturated outputs. In contrast, the adversarial loss encourages the generator to produce realistic colorizations, even if they deviate from the ground truth.

This difference is most apparent in areas where multiple colorizations would be plausible, such as clothing, vehicles, and manufactured items. While regression models typically produce grayish or brownish tones for these objects, our GAN model generates more definitive colors, producing more vibrant and visually appealing results.

The improved feature extraction capabilities of the ResNet backbone also contribute to better semantic understanding compared to our previous VGG16-based approach. This is evident in the model’s ability to consistently apply appropriate colors to everyday objects across different contexts.

D. Limitations

While our model is an improvement over existing methods, our work still has several limitations. First, due to our CPU-only training environment, we were significantly limited in our ability to experiment with larger models and datasets. A single training run of 70 epochs took approximately 96 hours, and so the scope of our hyperparameter optimization was greatly constrained. Had we obtained greater computational power, we could have utilized larger batch sizes and more data, likely resulting in a more robust model.

While the model performs well on common objects and scenes, it sometimes produces inconsistent colorization for less frequent items or unusual viewpoints. This reflects limitations in the training data diversity and could be addressed with a larger, more varied dataset.

Initially, we planned to include metrics like LPIPS and Delta E; however, these efforts were hindered by technical issues related to color space conversion and channel alignment. These metrics could help with interpreting the perceptual quality of the colorizations beyond pixel-level accuracy.

Finally, the model occasionally uses unrealistic colors in human faces (especially in situations with suboptimal lighting conditions or unclear facial features). This is a common challenge in colorization and supports the need for specialized attention to sensitive semantic regions.

VII. FUTURE WORK

Based on our findings and the limitations identified, we propose several directions for future research:

- *Enhanced Perceptual Evaluation:* Computing LPIPS and Delta E metrics would improve our ability to evaluate colorization quality, as it would focus on perceptual similarity instead of pixel-level accuracy. This would

require resolving color space conversion and channel alignment issues.

- *Perceptual Loss Integration:* Incorporating a perceptual loss based on pre-trained feature extractors could further improve the visual quality of colorizations by encouraging similarity in feature space rather than pixel space.
- *Hardware Acceleration:* Moving to GPU-based training would significantly accelerate experimentation, allowing for larger models, bigger datasets, and more extensive hyperparameter optimization.
- *Semantic Guidance:* Explicitly incorporating semantic segmentation information could improve colorization consistency for specific object categories, particularly those with strong color associations or those requiring special handling (e.g., human faces).
- *Temporal Extension:* Extending the approach to video colorization by incorporating temporal consistency constraints would broaden the applicability of our work.

VIII. CONCLUSION

In this paper, we presented a comprehensive approach to automatic image colorization using a GAN-based architecture that integrates a ResNet-based U-Net generator with a multi-scale PatchGAN discriminator, enhanced by a progressive three-phase training strategy. Our results demonstrate that this architecture effectively addresses common limitations in traditional regression-based colorization methods, particularly the tendency to generate desaturated and semantically ambiguous outputs.

Progressive training proved essential for achieving stable convergence and visually compelling results. By decoupling generator pretraining from adversarial learning and gradually introducing the GAN loss, our training strategy mitigated typical issues in GAN optimization, such as discriminator overpowering or mode collapse. The final balance of L1 reconstruction loss for structural fidelity and adversarial loss for perceptual realism enabled the model to produce accurate and diverse colorizations.

Quantitative evaluation showed competitive performance, with a final L1 loss of 0.24, PSNR near 7.2 dB, and SSIM reaching 0.23 on the test set. While these metrics alone are modest, qualitative assessment revealed that our GAN-based model produced significantly more vibrant and semantically consistent colorizations than our previous VGG16-based autoencoder. Improvements were obvious in regions with multiple plausible color mappings, where adversarial learning favored more realistic and saturated hues.

Image colorization remains a multimodal and ill-posed problem, especially when operating on unconstrained real-world datasets. Our work contributes a practical, extensible framework that leverages recent advances in GAN architecture, training curriculum, and stability engineering. Despite operating under hardware constraints with CPU-based training, the final model achieved notable performance. These results provide a strong foundation for future extensions such as

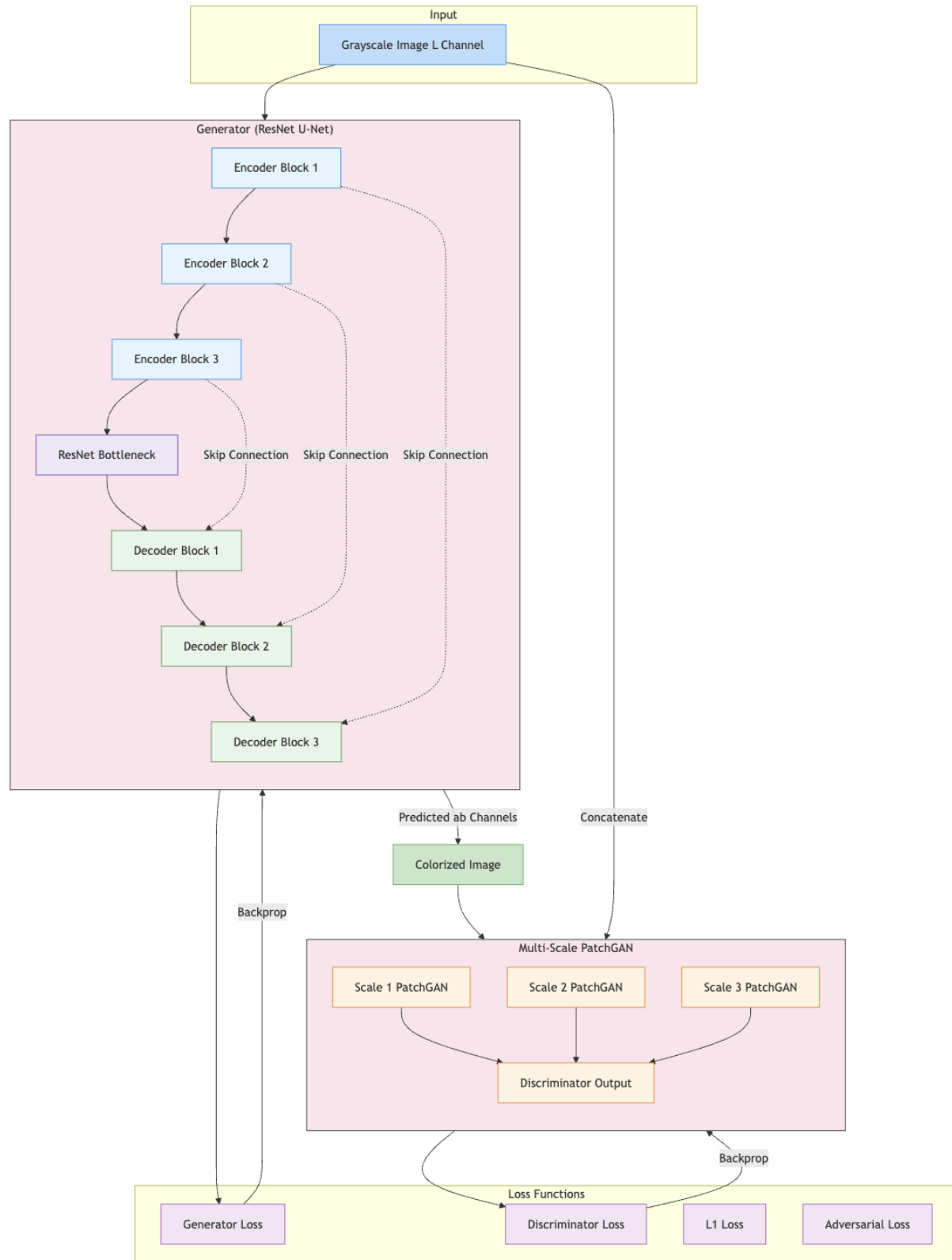
perceptual loss integration, semantic guidance, and real-time video colorization.

In sum, our findings reinforce the utility of progressive GAN training for image colorization and underscore the value of thoughtful architectural and optimization design in addressing multimodal generative tasks. This work not only demonstrates a functional GAN-based pipeline for colorization but also offers insights applicable to broader image-to-image translation problems.

REFERENCES

- [1] R. Zhang, P. Isola, and A. A. Efros, “Colorful Image Colorization,” Mar. 2016, doi: <https://doi.org/10.48550/arxiv.1603.08511>.
- [2] S. Iizuka, E. Simo-Serra, and H. Ishikawa, “Globally and locally consistent image completion,” *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–14, Jul. 2017, doi: <https://doi.org/10.1145/3072959.3073659>.
- [3] I. J. Goodfellow *et al.*, “Generative Adversarial Networks,” *arXiv (Cornell University)*, vol. 1, Jun. 2014, doi: <https://doi.org/10.48550/arxiv.1406.2661>.
- [4] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks,” Nov. 2016, doi: <https://doi.org/10.48550/arxiv.1611.07004>.
- [5] O. Bruvik and M. Penfold, “Image Colorization using GAN and Inception-ResNet-v2.”
- [6] Oliverio Theophilus Nathanael and Simeon Yuda Prasetyo, “Color and Attention for U: Modified Multi Attention U-Net for a Better Image Colorization,” *JOIV International Journal on Informatics Visualization*, vol. 8, no. 3, pp. 1453–1453, Sep. 2024, doi: <https://doi.org/10.62527/joiv.8.3.1828>.
- [7] Z. Du *et al.*, “Deeply supervised two stage generative adversarial network for stain normalization,” *Scientific Reports*, vol. 15, no. 1, Feb. 2025, doi: <https://doi.org/10.1038/s41598-025-91587-8>.

Fig. 4: Overall architecture of the proposed progressive GAN framework for image colorization. This figure outlines the data flow between the grayscale input, generator, multi-scale PatchGAN discriminator, and the loss functions.



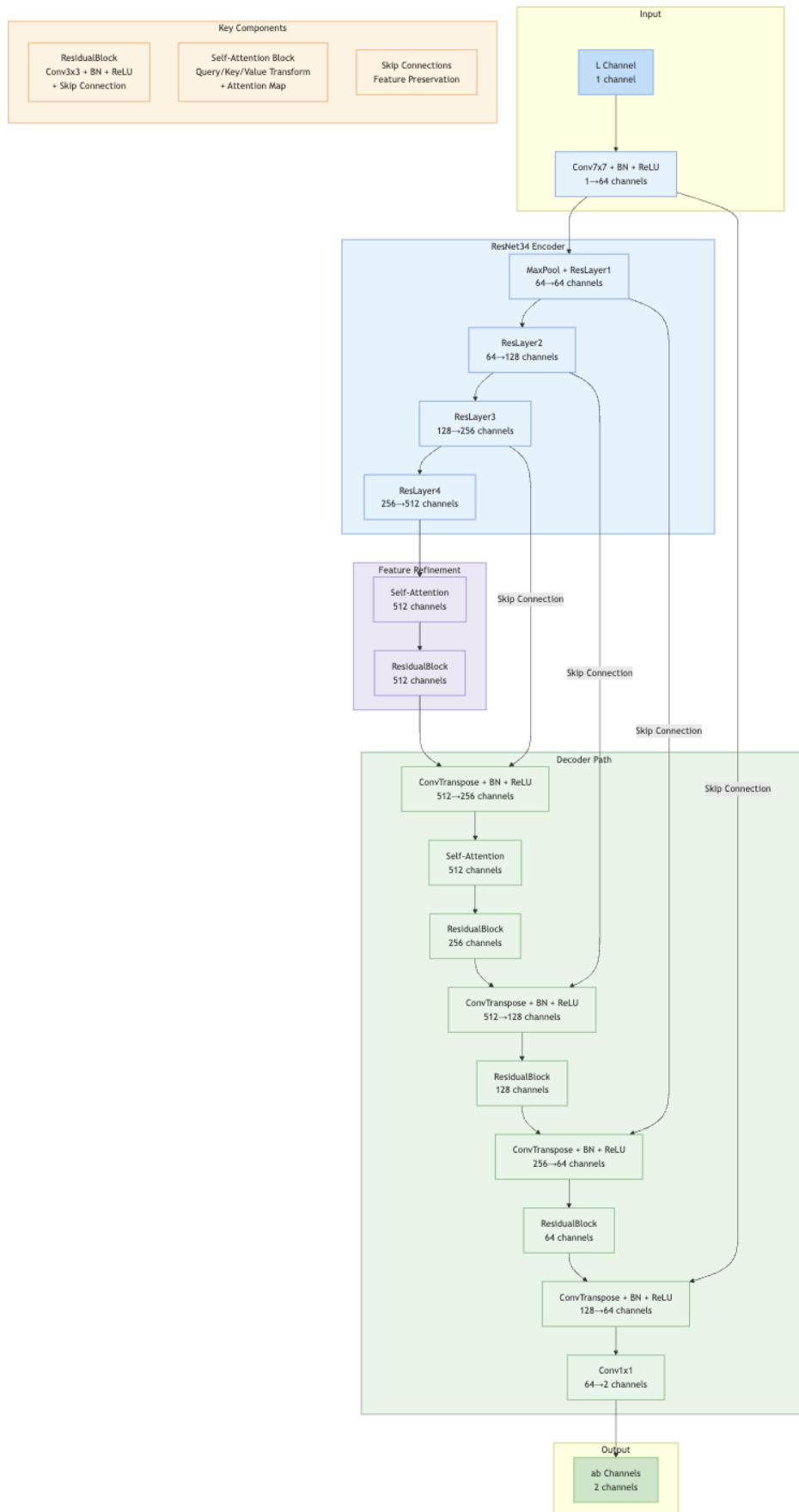


Fig. 5: ResNet-based U-Net generator with attention-enhanced skip connections.

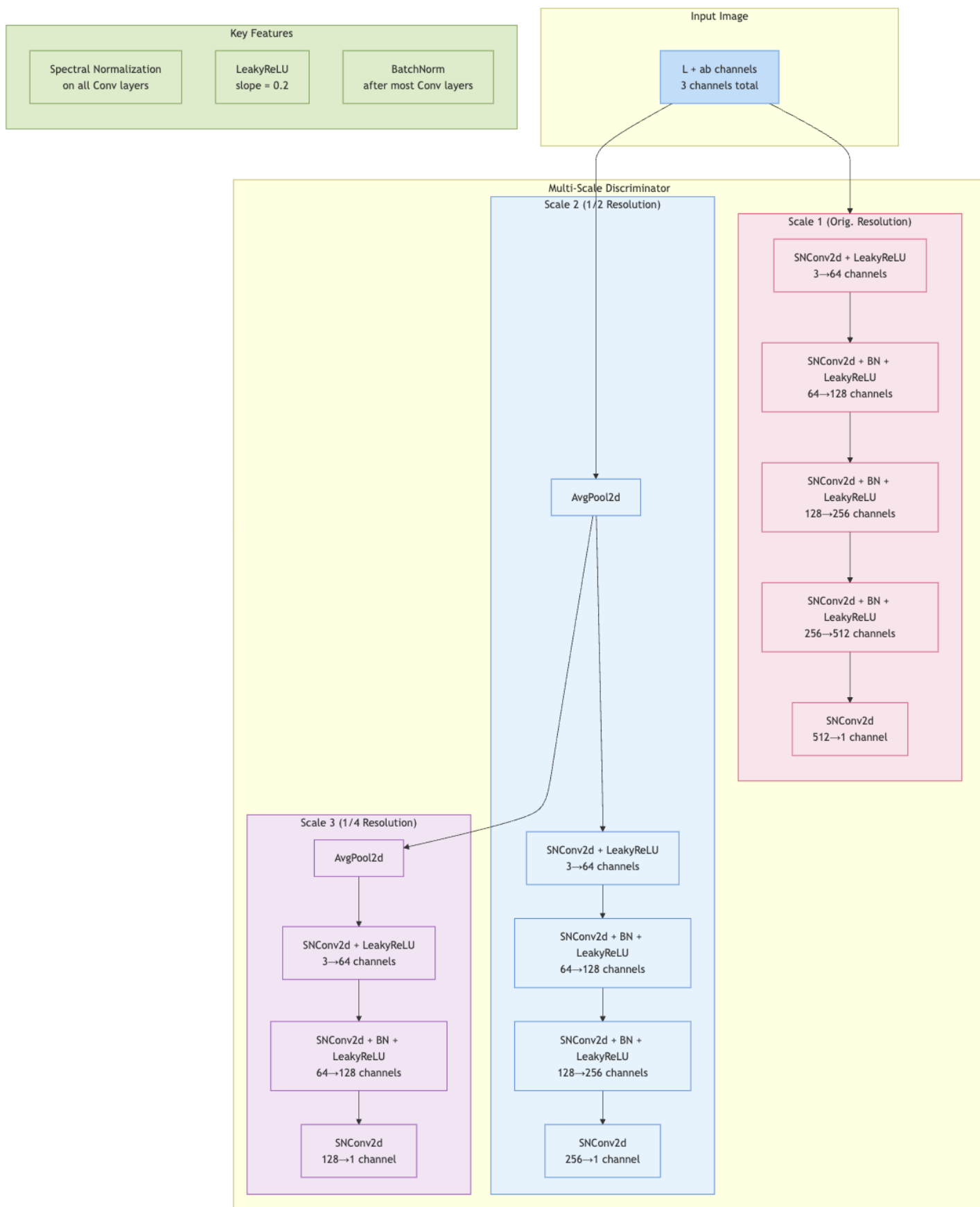


Fig. 6: Multi-scale PatchGAN discriminator architecture.