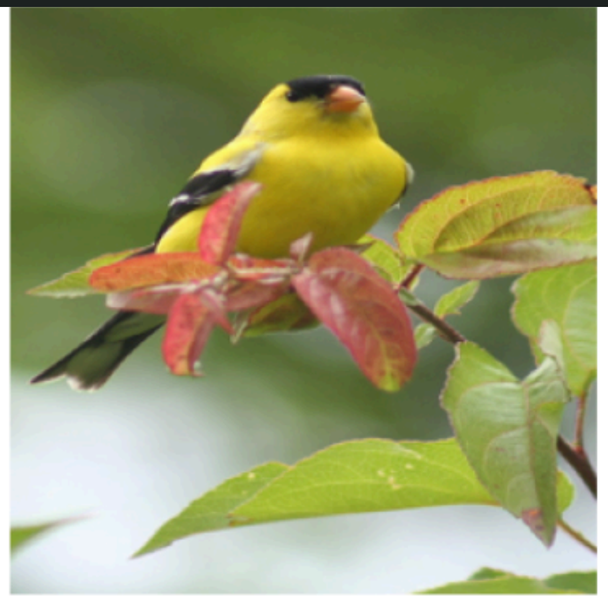
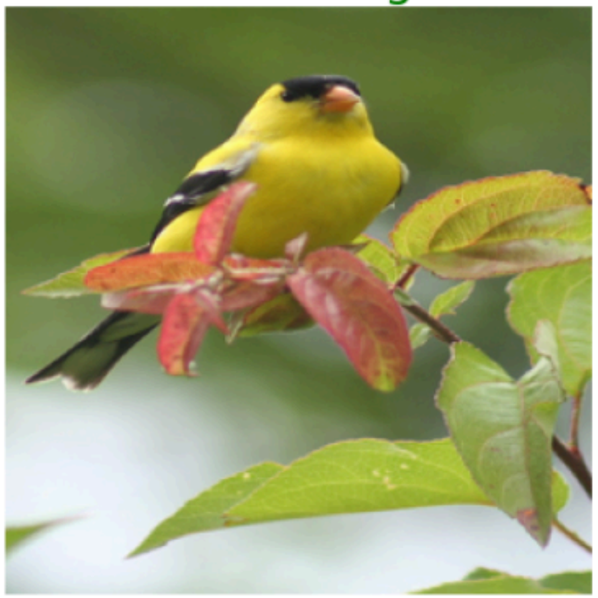
DEEP LEARNING PROJECT

AUTOMATIC COLOURISATION OF BLACK AND WHITE IMAGES

By: Devansh Upadhyaya, Dipit Golechha, Krishnav Mahansaria,
Vardan Vij and Yashvi Maheshwari



Color Image



Grayscale Image

MSE: 0.0102
PSNR: 19.91 dB
SSIM: 0.8597

INTRODUCTION

- Automatic colorization adds realistic colors to black-and-white images without human input
- It has applications in photo restoration, media enhancement, and forensics
- The task is challenging due to the ambiguity of multiple plausible colorisations for a single grayscale image
- Advanced techniques are essential for producing vibrant, consistent, and realistic results

OUR GOAL

- To develop a deep learning framework to colorise grayscale images effectively

- **Key Focus Areas —**

1. Realism — Produce visually vibrant and realistic colorisations
2. Accuracy — Ensure the output aligns with the underlying grayscale content

- **Challenges Addressed —**

1. Spatial consistency
2. Global context understanding
3. Semantic alignment of colours to objects

KEY METRICS

1. MSE (Mean Squared Error): Measures the average squared difference between predicted and ground truth pixel values; lower values indicate better reconstruction accuracy

2. PSNR (Peak Signal-to-Noise Ratio): Evaluates the quality of the reconstructed image by comparing it to the ground truth; higher values indicate better quality

3. SSIM (Structural Similarity Index): Assesses the perceived structural similarity between predicted and ground truth images; values close to 1 indicate higher similarity

4. FID (Frechet Inception Distance): Measures the distribution similarity between real and generated image features; lower values indicate closer alignment with real images

RELATED WORK

- Deep Learning Methods for Image Colorisation —

1. Convolutional Neural Network (CNN)-based models
2. Classification-based approaches
3. Generative Adversarial Network (GAN)-based models

- Each method has unique strengths but also faces limitations, providing opportunities for improvement

CNN BASED COLOURISATION

Reference: Sarapu, R., Viswanadam, A., Devulapally, S., Nenavath, H., & Ashwini, K. (2020). Automatic colorization of black and white images using convolutional neural networks. In Proceedings of the International Conference on Intelligent Computing and Control Systems.

- Method: Predict chrominance (a and b) channels in the CIE Lab color space from the luminance (L) channel using a Deep Learning Convolutional Neural Network (DLCNN)

- Results:

1. PSNR: **23.5** dB on the validation set.
2. Demonstrated reasonable pixel-level accuracy.

- Limitations:

1. Lacked vibrancy in outputs.
2. Struggled with complex scenes and distinguishing similar textures (e.g., grass vs. water)

COLORISATION AS A CLASSIFICATION TASK

Reference: Zhang, R., Isola, P., & Efros, A. A. (2016). Colorful image colorization. In European Conference on Computer Vision (pp. 649-666).

- Method:

1. Reframed colorsation as a classification problem.
2. Trained a deep CNN on a quantised colour space with a dataset of 1.3 million ImageNet images

- Results:

1. Top-1 Color Classification Accuracy: 65% on the quantized color space.
2. SSIM: 0.79, indicating improved perceptual quality

- Limitations:

1. Unrealistic results in complex scenes with overlapping objects due to a lack of global semantic understanding

GAN-BASED COLORISATION

Reference: Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1125-1134).

- Method:

1. Conditional GAN (cGAN) framework for image-to-image translation
2. Combined adversarial loss with pixel-wise reconstruction loss

- Results:

1. FID: 25.3, reflecting high perceptual quality
2. PSNR: 24.2 dB, indicating strong pixel-level accuracy

- Limitations:

1. Long-range dependencies were underutilised, causing spatial incoherence
2. Training instability required significant fine-tuning to prevent code collapse

OUR NOVELTIES

MULTISCALE INPUTS

- The generator processes the grayscale image at three scales:
  a. Original resolution.
  b. Downsampled (half resolution).
  c. Upsampled (double resolution).

- These inputs are concatenated to balance global context and fine-grained details

- Benefits:
  a. Captures both large-scale semantic information and local textures.
  b. Produces vibrant and consistent outputs

SELF-ATTENTION MECHANISMS

- Self-attention layers model relationships between distant regions of the image
- Example: Ensures colour continuity between the sky and its reflection in water
- Benefits:
  a. Enhances spatial coherence in complex scenes
  b. Captures dependencies across the image, improving realism

CONDITIONAL GAN (CGAN)

- Semantic features from a pretrained ResNet-18 are extracted and concatenated with the grayscale input for:

1. Generator: To guide color predictions based on image content
2. Discriminator: To assess alignment between colours and object semantics

- Benefits:
    a. Aligns generated colors with image semantics
    b. Reduces inconsistencies in outputs for complex scenes

IMPLEMENTATION

MODEL 1

- Architecture: Basic UNet paired with a PatchGAN discriminator
- Training Process:
  - Generator pre-trained with L1 Loss.
  - Full GAN trained using GAN Loss and L1 Loss for realism and pixel accuracy
- Output: Produces reasonable results but lacks spatial coherence and struggles with complex scenes

MODEL 2

- Key Enhancements:
- Inputs: Combines original, low-res, and high-res grayscale images
- Self-Attention: Improves spatial consistency and long-range dependencies
- Semantic Features: ResNet18 provides high-level context for both generator and discriminator
- Output: Delivers vibrant, context-aware colorizations with better performance on intricate images

1. DATASET PREPARATION

- Collected up to 10,000 images from ImageNet
- Split into 80% training and 20% validation sets
- Images resized to 256x256 pixels for uniformity
- Augmented the training dataset with random horizontal flips

2. DATA PRE-PROCESSING

- Images converted to Lab color space:
  - L channel (lightness): Used as input
  - ab channels (colour): Used as ground truth
- Normalised L channel to [-1, 1] and ab channels to [-1, 1].

3. MODEL ARCHITECTURE

- Generator:
  - Built using a basic UNet architecture
  - Predicts ab colour channels from L grayscale input
- Discriminator:
  - A simple PatchGAN architecture
  - Distinguishes real vs. fake colorisations based on local patches

4. LOSS FUNCTIONS

- GAN Loss: Encourages the generator to produce realistic images
- L1 Loss: Ensures pixel-level similarity between generated and ground truth ab channels

5. TRAINING WORKFLOW

- Step 1:
  - Pre-train the generator with only L1 loss for stabilization
- Step 2:
  - Train the generator and discriminator alternately:
  - The generator tries to fool the discriminator
  - The discriminator learns to distinguish real from fake
- Step 3:
  - Save model weights and log training losses for analysis

MODEL 1

1.DATASET PREPARATION

- Follows the same process as the first model —
  - Collected 10,000 images.
  - Split into 80% training and 20% validation sets.
  - Images resized to 256x256 pixels with random horizontal flips

2.DATA PRE-PROCESSING

- Converted images to Lab colour space —
  - L channel: Grayscale input, normalised to [-1, 1]
  - ab channels: Color output, normalized to [-1, 1]

3.MODEL ARCHITECTURE

- Generator:
  - Based on UNet with key enhancements:
  - Multiscale Inputs — Original, low-res, and high-res versions of the L channel
  - Self-Attention Layer — Ensures spatial coherence
  - ResNet18 Semantic Features — Provides high-level context
- Discriminator:
  - Conditional PatchGAN: Enhanced with ResNet18 features for semantic conditioning

4. LOSS FUNCTIONS

- GAN Loss: Similar to the first model
- L1 Loss: Maintains pixel-level accuracy
- Improvement: Semantic features and multiscale inputs boost alignment with ground truth

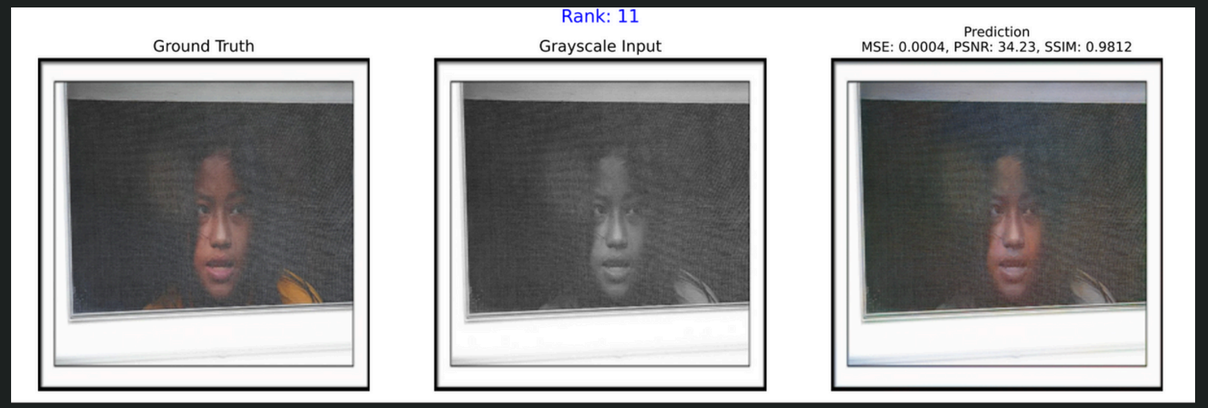
5. TRAINING WORKFLOW

- Step 1
  - Pre-train the generator using only L1 loss
- Step 2
  - Train the full Conditional GAN:
  - Generator optimised to minimise L1 and GAN losses
  - Discriminator learns to differentiate real from fake based on patches and semantic context
- Step 3
  - Save model weights and log training details

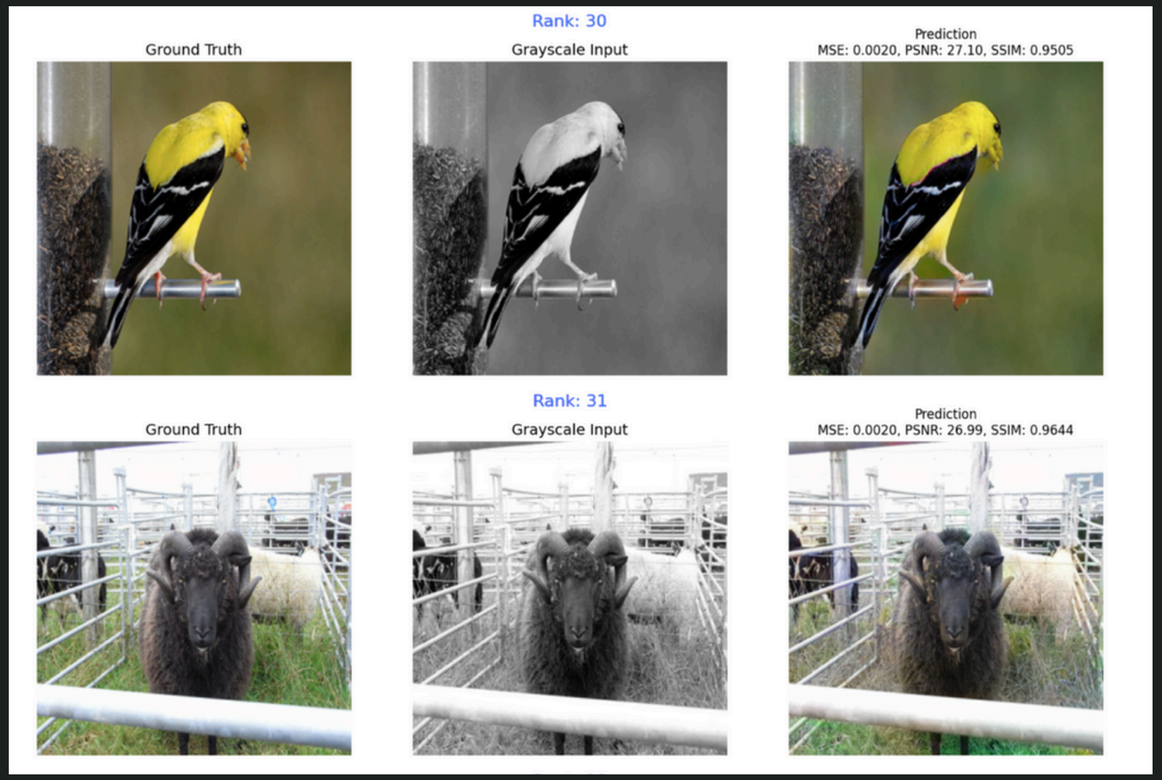
MODEL 2

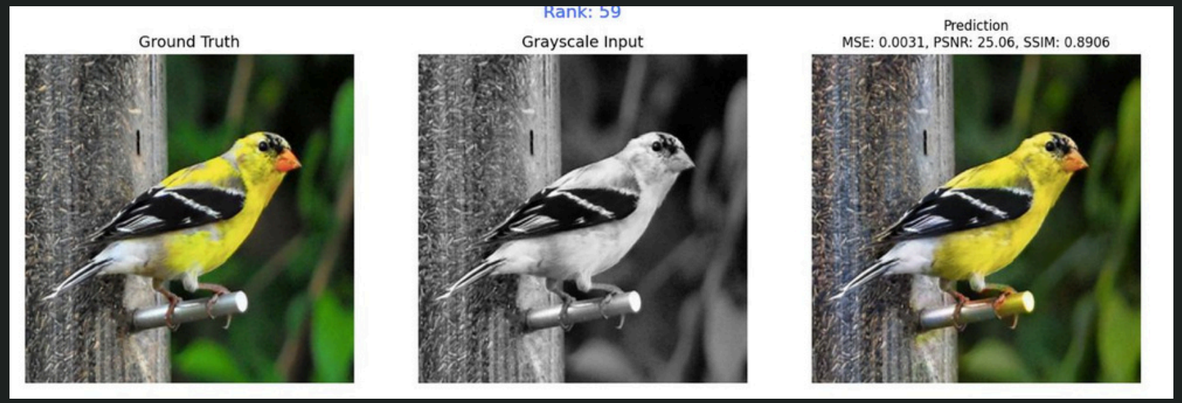
Feature	First Model	Second Model
Generator	Basic UNet	**UNet with multiscale inputs and semantic feature**
Discriminator	Patch GAN	**Conditional Patch GAN with ResNet18 conditioning**
Pre-processing	Standard Resizing and Lab Conversion	**Identical Pre-processing**
Loss Functions	GAN Loss + L1 Loss	**GAN Loss + L1 Loss with improved generator and discriminator interaction.**
Visualisations	Grey Scale, Generated and Actual Images	**Same with improved outputs**

RESULTS - MODEL 1

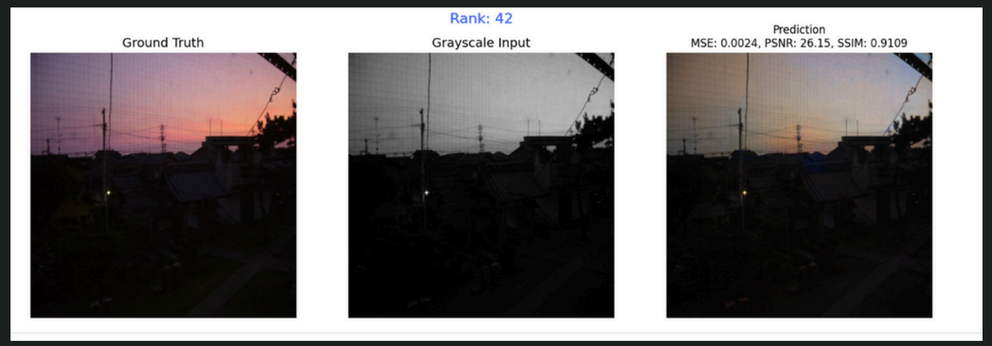


RESULTS - MODEL 2





RESULTS NOT MATCHING



LOSSES

- loss D fake and loss D real: These losses represent how well the discriminator

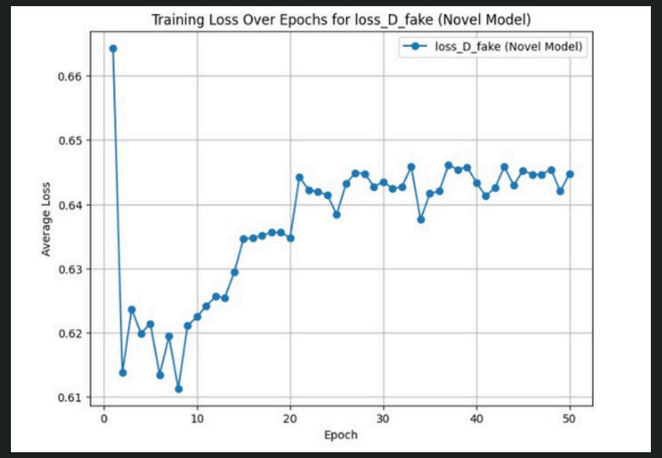
distinguishes between fake and real images.

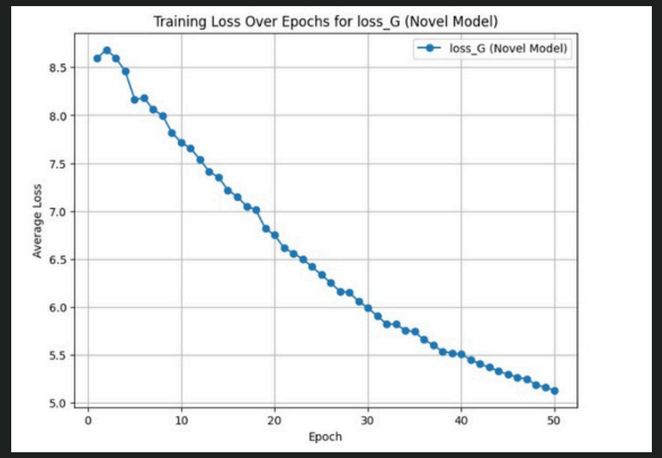
- loss D fake: Measures how confidently the discriminator identifies generated images as fake.
- loss D real: Measures how confidently the discriminator identifies real images as real.
- loss D: The average of loss D fake and loss D real. It represents the overall loss for the discriminator.

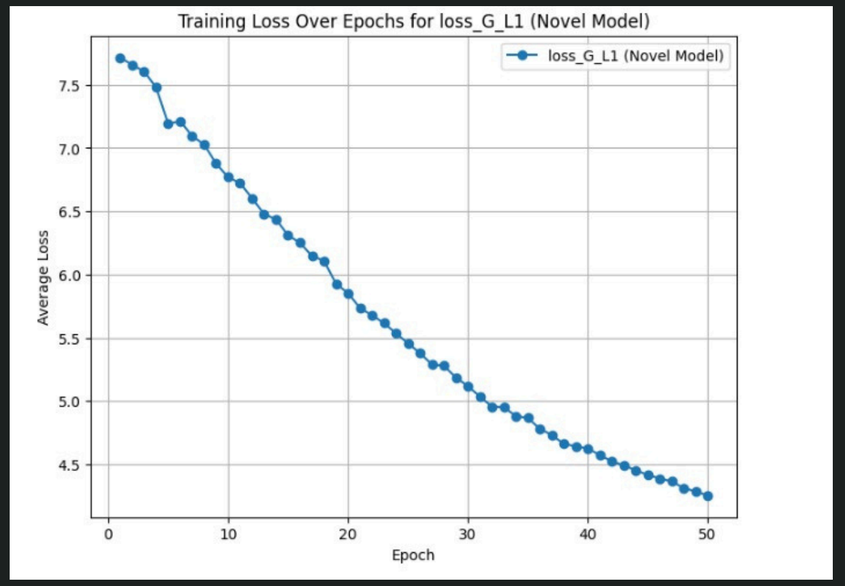
LOSSES

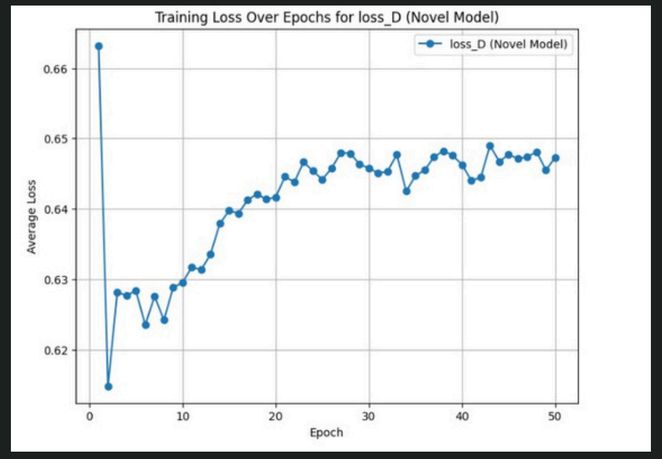
- loss G L1: This is the L1 loss between the generated and target (real) images, encouraging the generator to produce outputs close to the real data.
- loss G: The combined generator loss (loss G GAN + weighted loss G L1).
- loss G GAN: This measures how well the generator is fooling the discriminator.

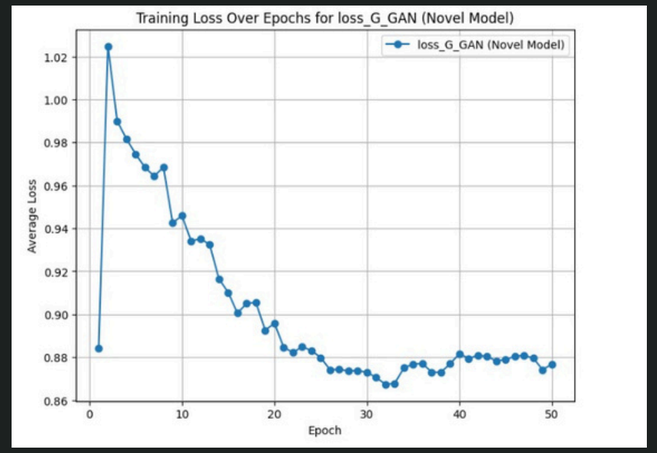
TRAINING RESULTS



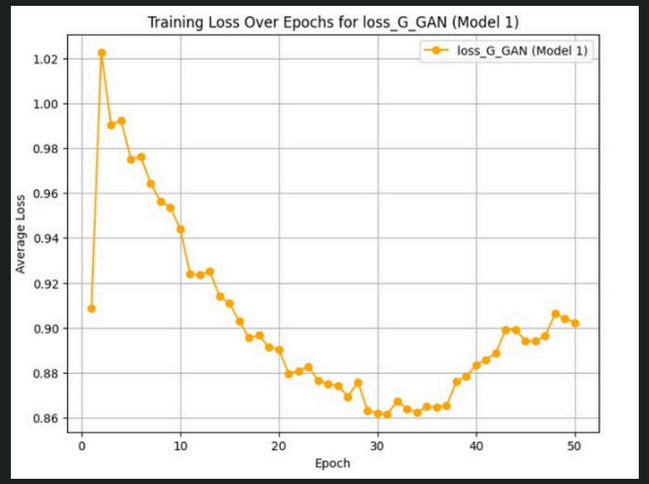


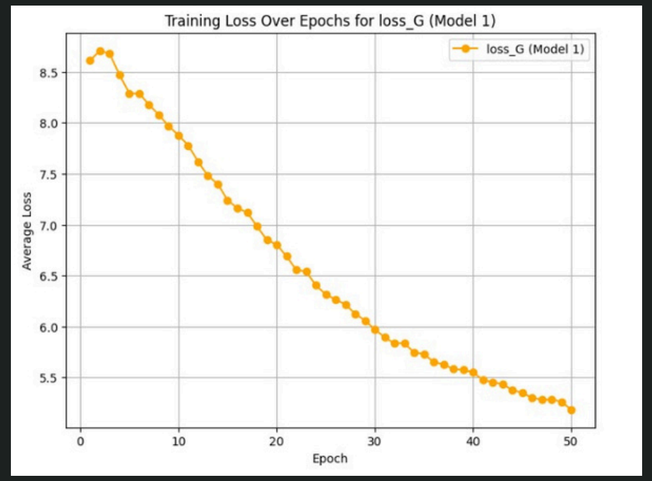


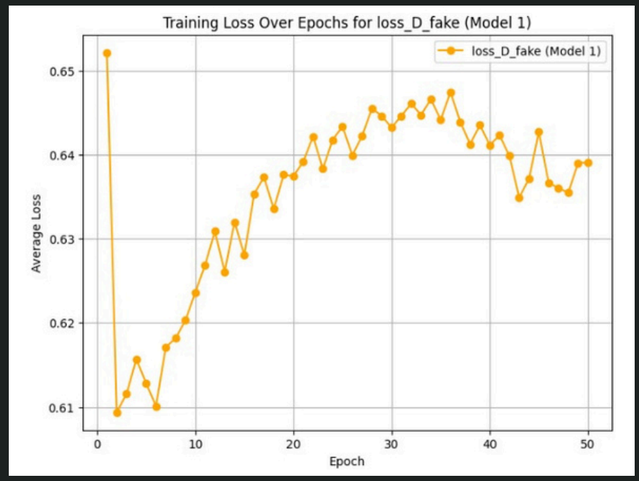


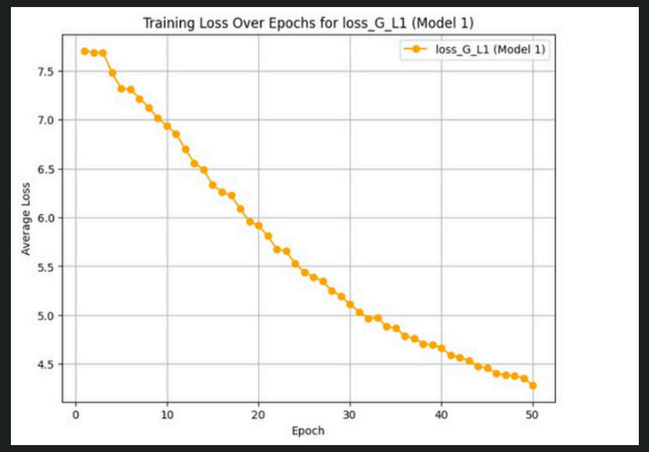


RESULTS


Training Loss Over Epochs for loss_G_GAN (Model 1)
loss_G_GAN (Model 1)
Average Loss
Epoch


Training Loss Over Epochs for loss_G (Model 1)
loss_G (Model 1)
Average Loss
Epoch


Training Loss Over Epochs for loss_D_fake (Model 1)
loss_D_fake (Model 1)
Average Loss
Epoch


Training Loss Over Epochs for loss_G_L1 (Model 1)
loss_G_L1 (Model 1)
Average Loss
Epoch


Training Loss Over Epochs for loss_D (Model 1)
loss_D (Model 1)
Average Loss
Epoch

RESULTS

MODELS	AVERAGE METRICS			
	MSE	PSNR	SSIM	FID
MODEL 1	0.0060	23.78 DB	0.9108	34.4180
MODEL 2	**0.0053**	**23.85 DB**	**0.9224**	**36.334**

LIMITATIONS

- Limited dataset of 8000 images for training, and 2000 for validation due to computational resource constraints
- Pre-trained model used for semantic features, introducing compounding errors due to lack of semantic labels in ImageNet
- Model trained for only 50 epochs, with the generator pre-trained for 20 epochs, due to computational resource limits
- MSE, PSNR, and SSIM do not capture human perception of colour quality and realism, making model assessment less intuitive

CONCLUSION

Results

1. Model 2:
    - Lower MSE: 0.0053 vs 0.0060
    - Higher PSNR: 23.85 dB vs 23.78 dB
    - Better SSIM: 0.9224 vs 0.9108
2. Slightly higher FID: 36.334 vs 34.4180

Recommendation

Model 2 preferred for high-quality, stable, and contextually coherent image colorisation

REFERENCES

- P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1125–1134, 2017.
- R. Sarapu, A. Viswanadam, S. Devulapally, H. Nenavath, and K. Ashwini. Automatic colorization of black and white images using convolutional neural networks. In Proceedings of the International Conference on Intelligent Computing and Control Systems, 2020.
- R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In European Conference on Computer Vision, pages 649–666, 2016.

THANK YOU!