“Colorization of grayscale images using Deep Neural Networks”

Deep Learning DATS_6303

Group 1:

Naiska Buyandalai
Sai Rachana Kandikattu
Snehitha Tadapaneni

→

If I show you a black-and-white photo, how many ‘correct’ color versions exist???

→

Not just guessed but algorithmically hallucinated???

Motivation

The challenge with image colorization is that one grayscale input corresponds to infinitely many possible color outputs.

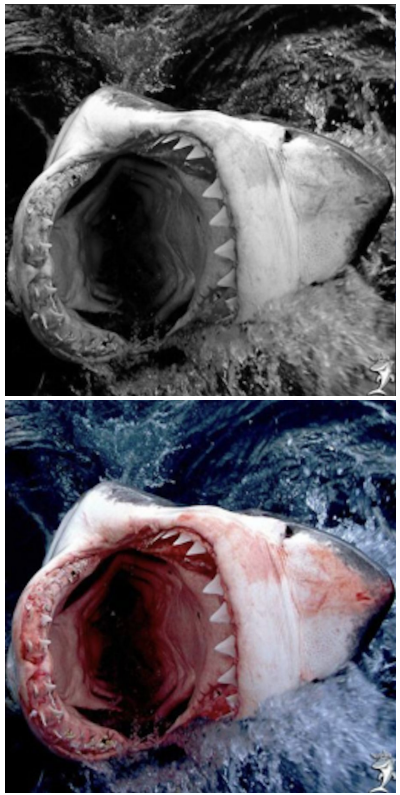
Our project explores how a model learns to make the *most plausible* choice.

- We build on the foundational ECCV16 work by Zhang, Isola, and Efros, which reframed colorization as a classification problem. We use their pretrained model as our baseline and extend it with a GAN-based architecture.

- Based on Zhang et al., ECCV 2016 ("Colorful Image Colorization")

Research Goal

1. Reproduce and evaluate the pretrained ECCV16 model by Zhang et al.
2. Develop an enhanced colorization model using GAN.
3. Compare both models using perceptual and quantitative metrics.
4. Analyze improvements in realism, saturation, and structural consistency.



Result from our enhanced GAN

Dataset Overview : ImageNet-50 Subset

We used the ImageNet-50 subset available

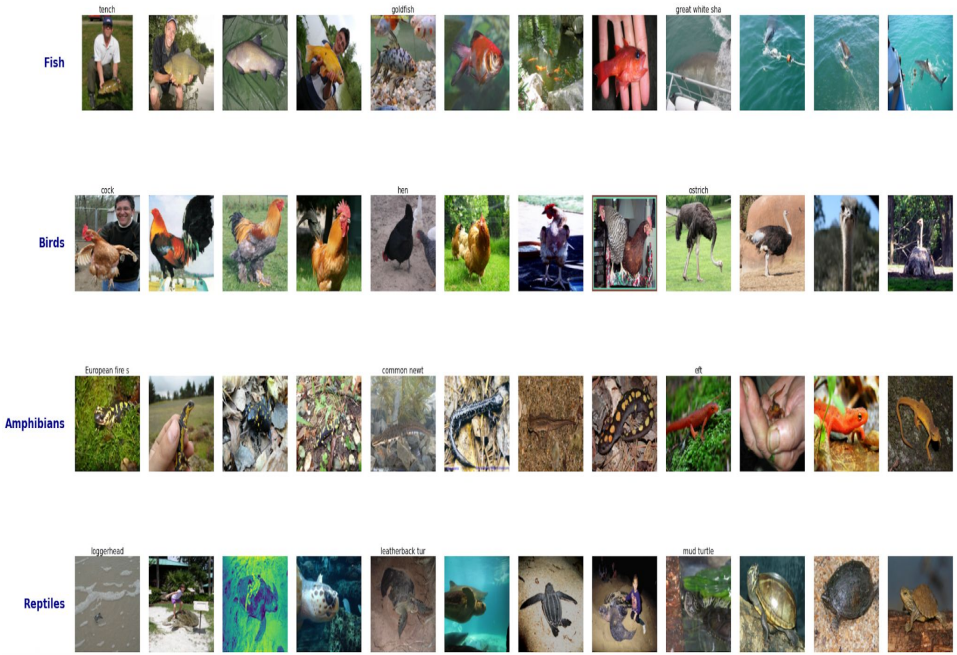
on HuggingFace.

It contains:

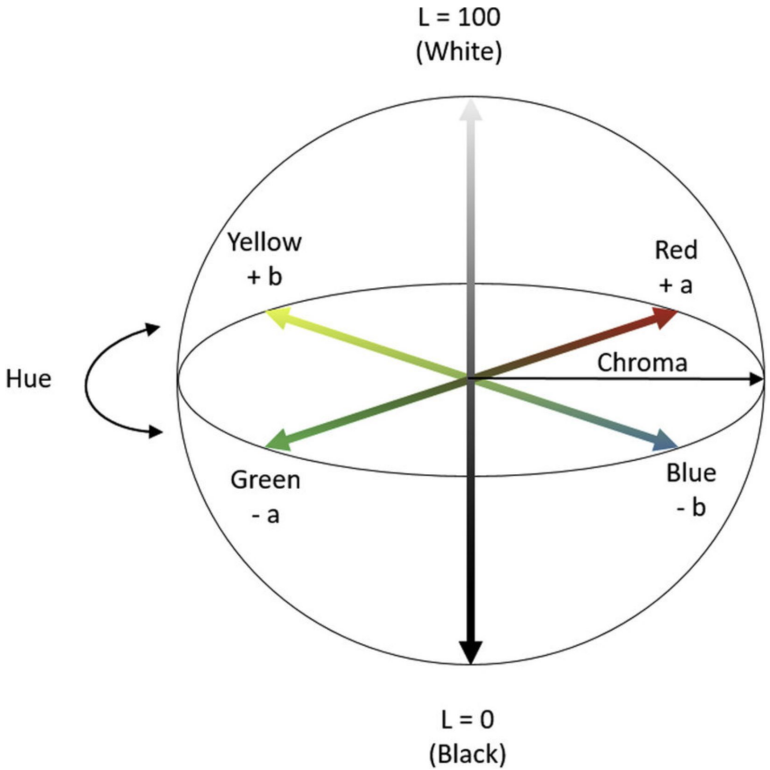
- 50 classes
- ~50,000 color images
- ~1,000 images per class

All images are high-quality natural scenes, objects, and animals.

The dataset provides diverse color distributions important for robust colorization.



CIELAB Color Space



➔ A perceptually uniform color space with three channels:
L = Lightness (0–100)
a = Green ↔ Red
b = Blue ↔ Yellow

➔ Why LAB instead of RGB?

In LAB, Color (a,b) is separated from structure (L)
So, the model will learn colorization as
Given (L) -> Predict (a,b)

RGB mixes brightness with color which makes prediction unstable.

→

Ly, Bao & Dyer, Ethan & Feig, Jessica & Chien, Anna & Bino, Sandra. (2020).

Methodology

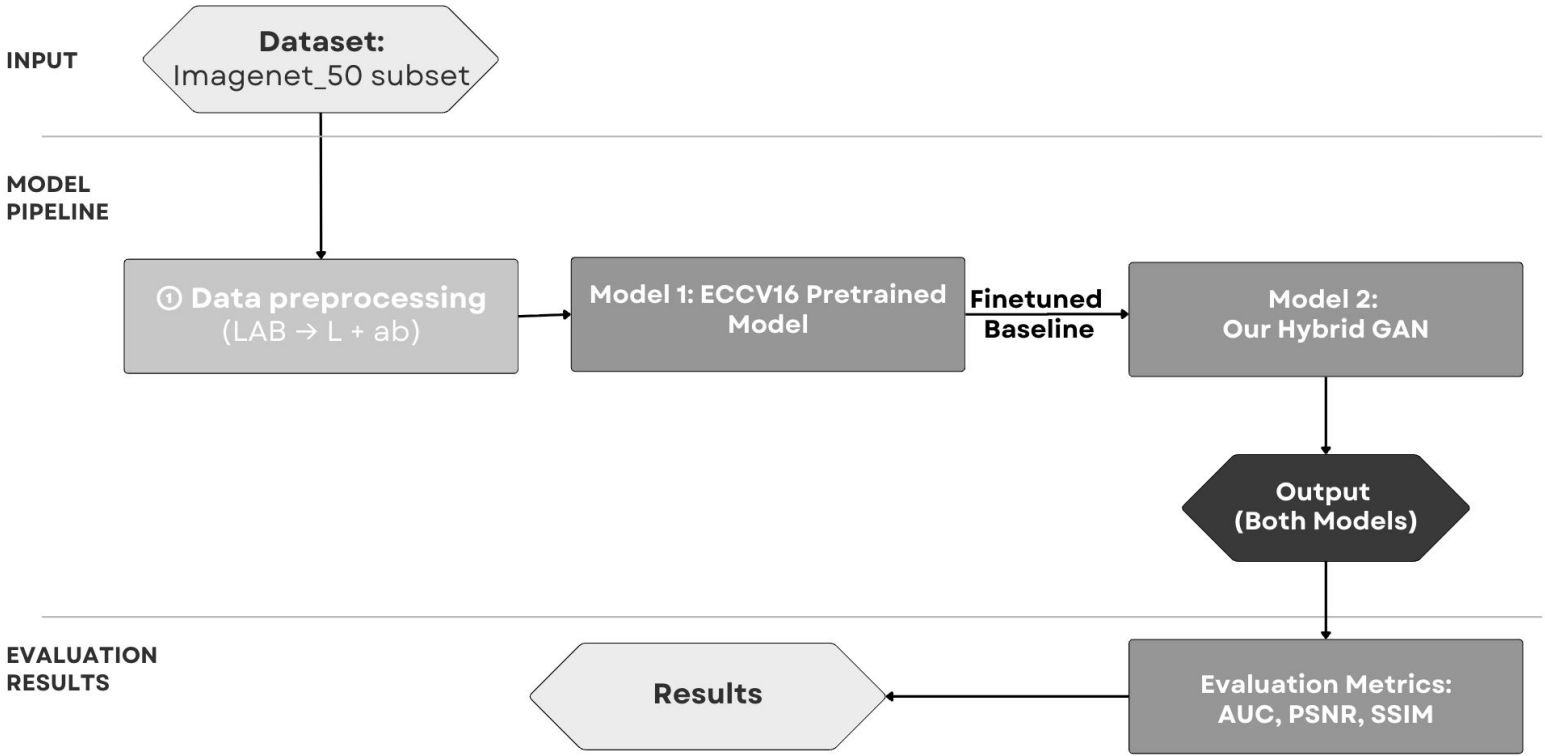


Figure: Overview of our colorization pipeline showing preprocessing, baseline model, GAN model, and evaluation.

Data Preprocessing

1. **Load RGB Images**

   Images from the ImageNet-50 subset (train + val).

2. **Resize to 256×256**

   Ensures consistent spatial resolution for both models.

3. **Convert RGB → CIELAB Space**

   LAB separates luminance (L) from chrominance (a,b).

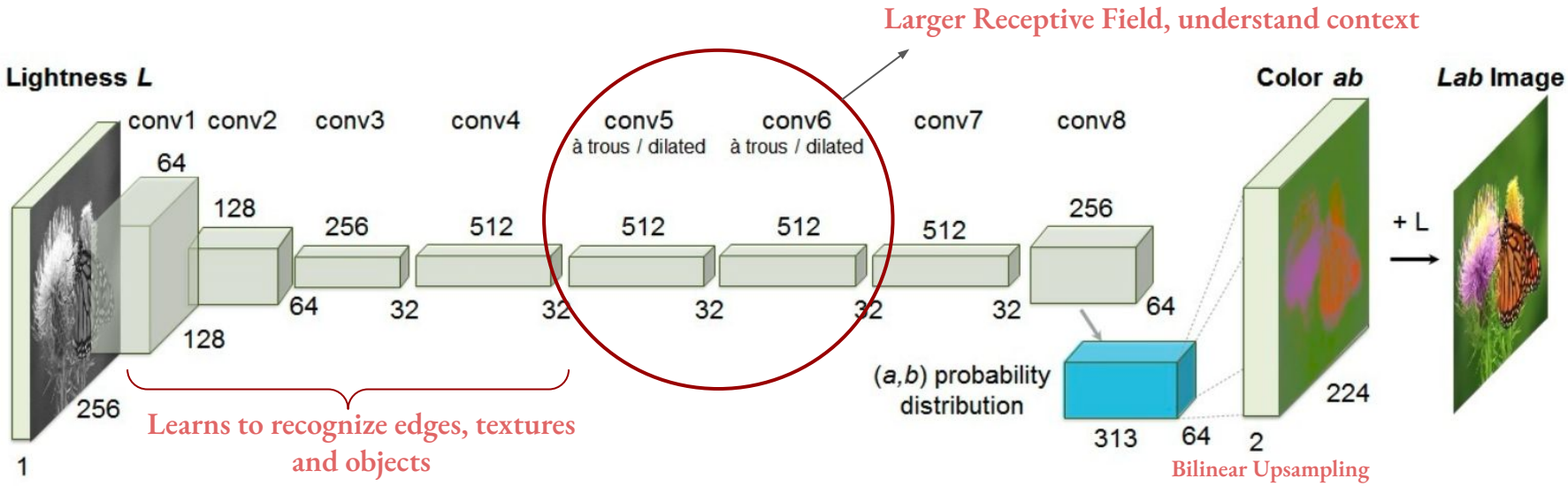
4. **Extract Channels for Model Input**

   L channel → input to both models | ab channels → ground truth for supervised training

5. **Normalize L and ab Channels**

   Follows Zhang et al.'s ECCV16 normalization
   (centered L at 50, scaled by 100; ab scaled by 110).

Pretrained Model: ECCV16 Architecture



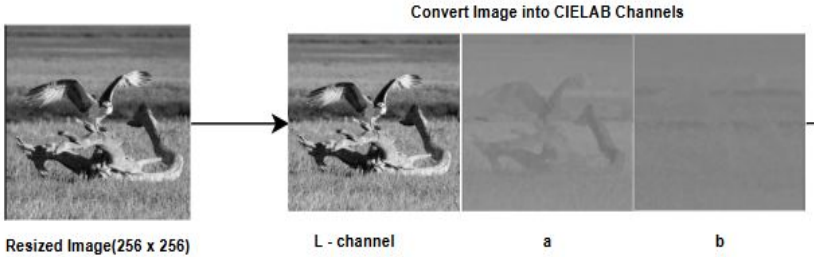
Zhang, R., Isola, P., & Efros, A. (2016), "Colorful Image Colorization," *arXiv preprint arXiv:1603.08511*.

Data Post Processing

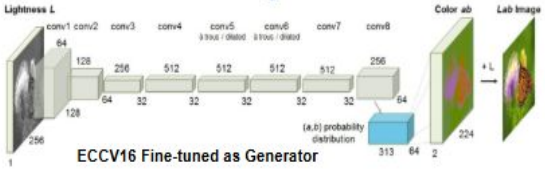
1. **Model predicts ab channels**
   Output is typically lower resolution (e.g., 56×56 from ECCV16).
2. **Upsample the predicted ab**
   Apply bilinear interpolation to match the original L-channel size.
3. **Concatenate with original L**
   Combine:
   L_original ab_predicted_upsampled

4. **Convert back to RGB**
   a. Use LAB → RGB transformation.
   b. Produces the final viewable color image.

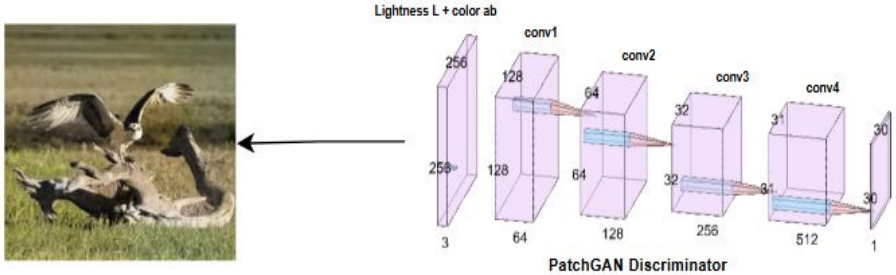
GAN Architecture:



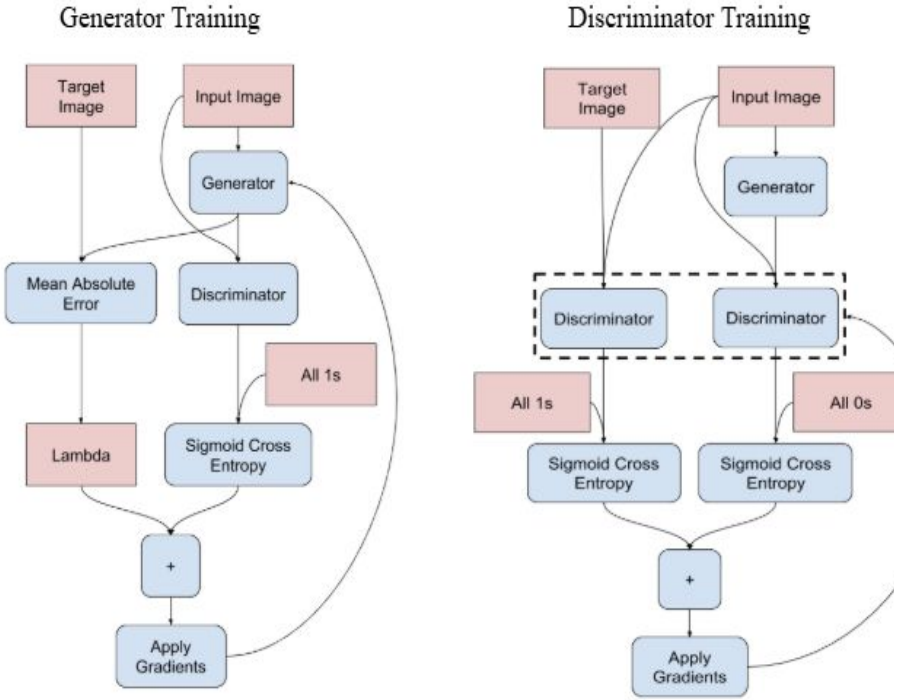
ECCV16 internally normalises input and unnormalises output



Output of Generator normalised to [-1,1] for discriminator



GAN Training:



$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_{x,y}\left[\log D(x,y)\right] + \mathbb{E}_{x,z}\left[\log(1 - D(x, G(x,z)))\right]$$

$$G^* = \arg\min_G \arg\max_D \left[\mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G)\right]$$

Two-Stage Training:

1. Stage 1: Finetune ECCV16 with MSE loss (3 epochs)
2. Stage 2: GAN training with adversarial + L1 + perceptual loss (50 epochs)

Loss Functions:

Generator Loss = GAN Loss + λ·L1 Loss + Perceptual Loss

↓ (fool D) ↓ (λ=10) ↓ (VGG19 features)

Discriminator Loss = 0.5·(Loss_real + Loss_fake)

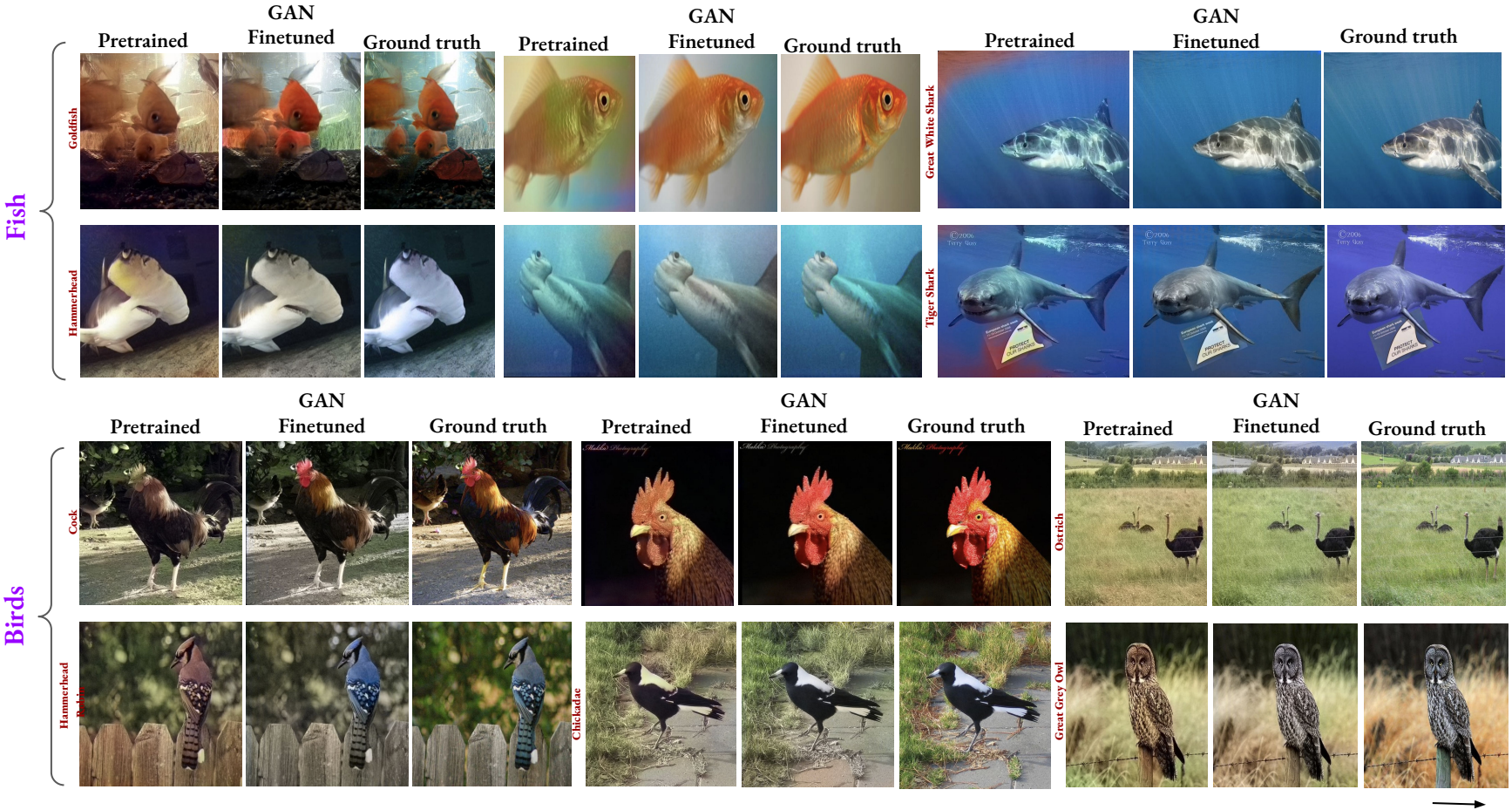
Discriminator: Maximize detection accuracy → Tell real from fake

Generator: Minimize detection accuracy → Make fakes ~~look real~~

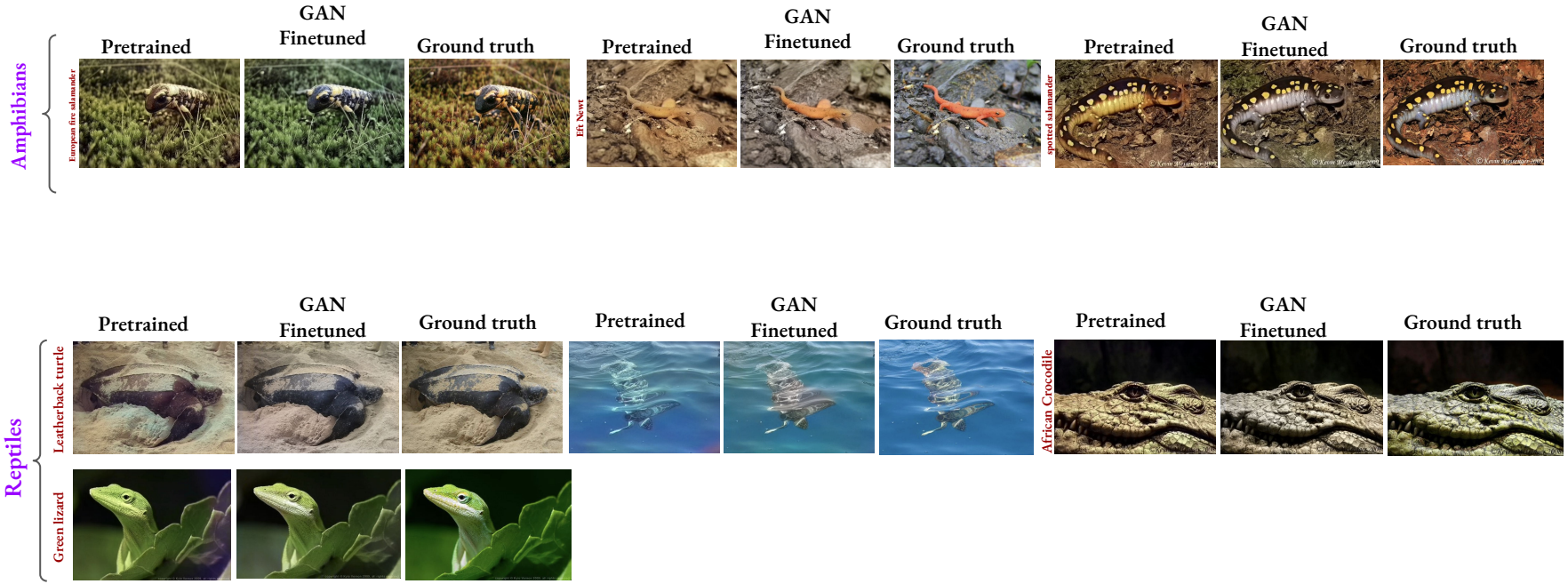
Parameters

Component	Value	Experimented	Rationale
Learning Rates	G: 0.0002 D: 0.0001	Experimented with (D -> 0.0002) [Standard pix2pix]	Slower D prevents overpowering G
Epochs	Finetune: 3 GAN: 50	Experimented with {5, 10, 25, 30} epochs	Preserve pretrained knowledge Allow adversarial convergence
Batch Size	16	Experimented with {1, 4, 16} for GAN	Pix2Pix suggests 1 but 16 provided the best results
Lambda (λ)	10	Tested {1, 10, 100} → 10 optimal	Better PSNR
Image Size	256×256	Match ECCV16 input	Standard
Optimizer	Adam	(β_1=0.5, β_2=0.999)Standard for GANs	Standard

Success cases



Success cases



Failure cases

Failure cases

Amphibians

Axolotl

Pretrained | GAN Finetuned | Ground truth

Bull Frog

Pretrained | GAN Finetuned | Ground truth

Reptiles

Leatherback turtle

Pretrained | GAN Finetuned | Ground truth

Pretrained | GAN Finetuned | Ground truth

Evaluation Metrics….

Comparison: Pretrained vs Enhanced GAN

Evaluation Metric	AUC	PSNR	SSIM
Pretrained (Zhang et al.)	0.89	22.27	0.91
Enhanced GAN	0.91	23.76	0.92

Experimental Game

Which one do you think is the original????

Image 1

Which one do you think is the original????

Image 2

Which one do you think is the original????

Image 1

Predicted Image

Image 2

Ground truth Image

Which one do you think is the original????

Image 1

Which one do you think is the original????

Image 2

Which one do you think is the original????

Image 1

Image 2

Predicted Image

Ground truth Image

Conclusion

- Enhanced GAN model improved realism and perceptual quality compared to the ECCV-16 base model
- Quantitative metrics (AUC, PSNR, SSIM) showed GAN model is outperforming the base model
- Visual inspection confirmed more natural colors and better contrast in several samples

Limitation

- Performance varies across categories
- Generalization limited to ImageNet like natural images
- Ambiguity in “correct” color remains

Streamlit

LINK: Streamlit App