Polygon Coloring with Conditional UNet

Aishwarya K P

August 5, 2025

1 Problem Statement

This project implements a **Conditional UNet** to fill a polygon outline image with a specified color. The model takes two inputs:

1. Polygon outline image (e.g., triangle, square, octagon)
2. Color name (e.g., red, blue, yellow)

Output: An RGB image of the same polygon filled with the given color.

2 Dataset

The dataset consists of paired inputs (polygon + color) and outputs (colored polygon).

Folder structure:

```
dataset/
  training/
    inputs/      # Polygon outline images
    outputs/     # Corresponding colored polygons
    data.json    # Mapping polygon image + color → output image
  validation/
    inputs/
    outputs/
    data.json
```

Note: Dataset JSON key `"colour"` (British spelling) was handled in preprocessing.

3 Model Architecture

- **Base model:** UNet
- **Conditioning:**
  - One-hot encode color
  - Pass through a Linear layer to produce a 64-dimensional embedding
  - Expand embedding to a $(64 \times H \times W)$ tensor and concatenate with polygon image along the channel dimension
- **UNet Encoder–Decoder:**
  - Downsampling path: 3 convolutional blocks with MaxPooling
  - Bottleneck: 512 channels
  - Upsampling path: transposed convolutions with skip connections
- **Output:** 3-channel RGB image with sigmoid activation

4 Hyperparameters

Parameter	Value
Image Size	128×128
Batch Size	8
Learning Rate	1×10^{-4}
Epochs	30
Loss Function	L1 Loss

5 Training Details

- **Framework:** PyTorch
- **Experiment Tracking:** Weights & Biases
- **GPU Used:** None (CPU Training on local machine)

W&B run link: `https://wandb.ai/kpaishwarya001-ayna/polygon-coloring`

5.1 Training and Validation Loss Curves

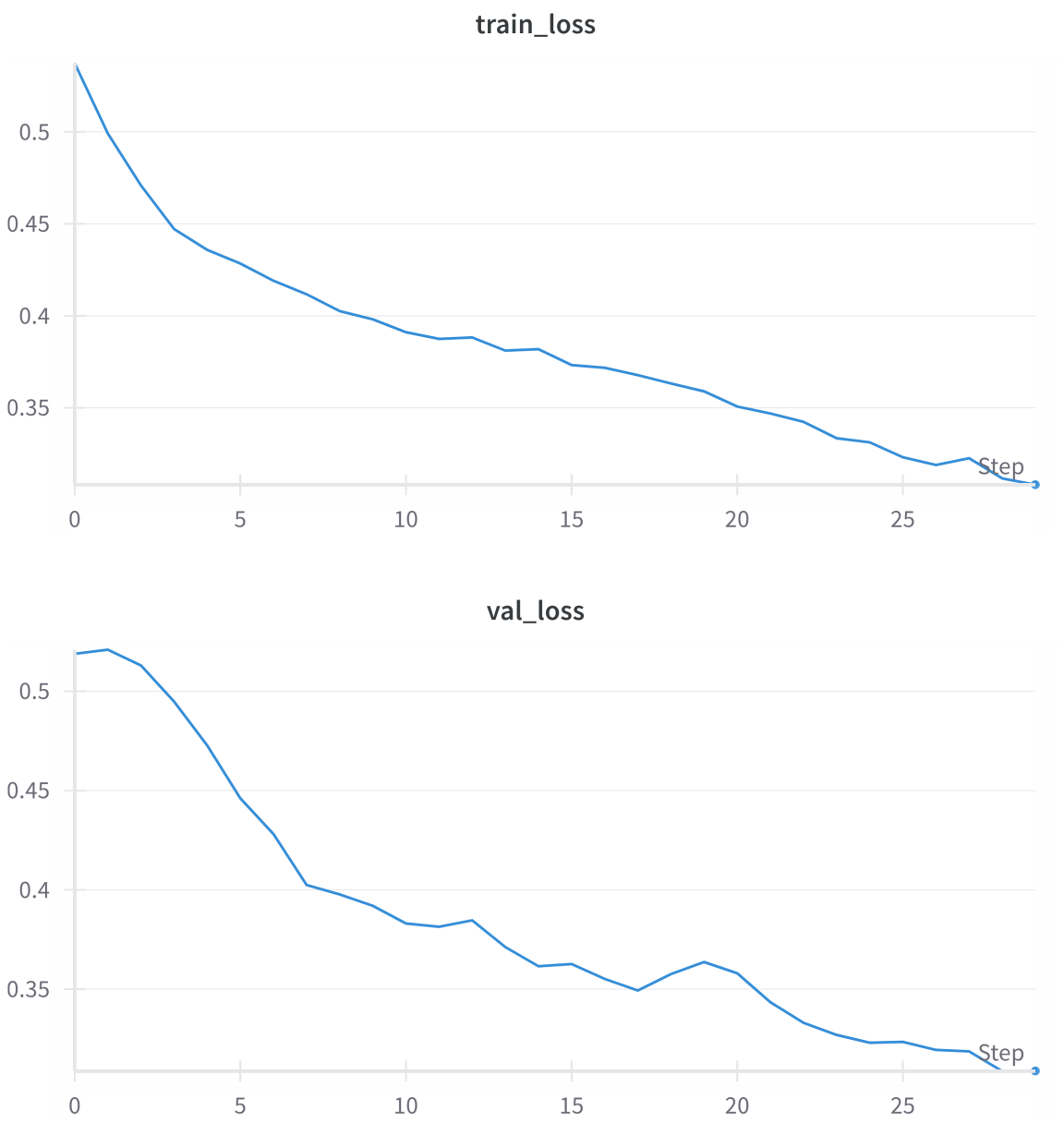

Figure 1: Training and Validation Loss Curves from W&B

6 Example Predictions

Below are sample results from the trained model. Each example shows the input polygon (top row) and the predicted colored polygon (bottom row).

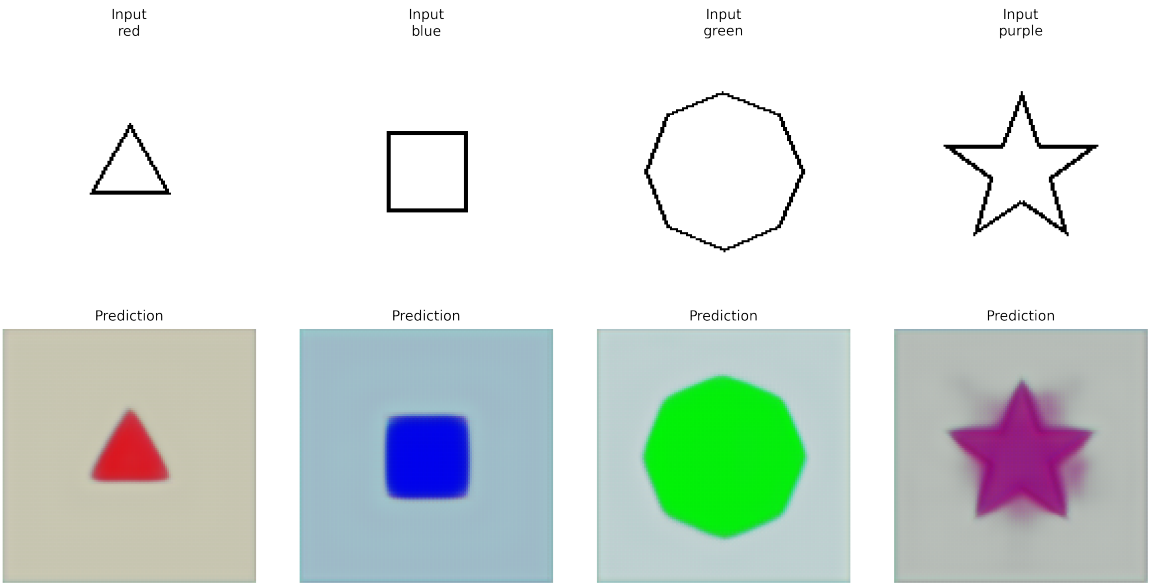

Figure 2: Example Predictions

7 Challenges & Fixes

- **Dataset key mismatch:** The JSON file used `"colour"` instead of `"color"` → fixed in dataset loader.
- **Shape mismatches:** Ensured train and validation datasets share the same n_colors.
- **Color conditioning issues:** Fixed embedding expansion to match UNet input dimensions.

8 Key Learnings

- Implementing Conditional UNet from scratch in PyTorch
- Handling multi-input models (image + categorical input)
- Using W&B for experiment tracking
- Debugging shape mismatches and aligning dataset mappings