SKETCHBOT

A Real-Time Application for Classification & Generation

School of Mathematics & Statistics

Aakriti Abhay Singh 24215296

Charukeshi Senthamarai Kannan 24213384

1. INTRODUCTION

We present **SketchBot**, a real-time **React + Flask** system for sketch **recognition** and **generation** across 10 Google **Quick, Draw!** classes.

The pipeline enforces identical train serve preprocessing (binarize → crop → pad → resize 64×64 → normalize) to avoid distribution shift.

- We use a compact **SketchCNN** that takes the 64×64 grayscale canvas and outputs 10-way class probabilities, enabling real-time recognition of what the user is drawing.
- We adopt a **Sketch-RNN–style** conditional VAE (**GenerateRNN**): a **BiLSTM encoder** maps stroke sequences to a latent code, and an **LSTM + MDN decoder** samples pen states to generate sketches of classes using a generative deep-learning model.

Both services are exposed through simple HTTP endpoints (/predict, /category) for interactive use. We evaluate the system on the 10-class benchmark and demonstrate robust classification and coherent, class-conditioned drawings in real time

Bat	Bicycle	Bus	Cactus	Clock
Door	Guitar	Lightbulb	Paintbrush	smileyface

Table 1. QuickDraw classes used for this work

2. MODEL ARCHITECTURE & PIPELINES

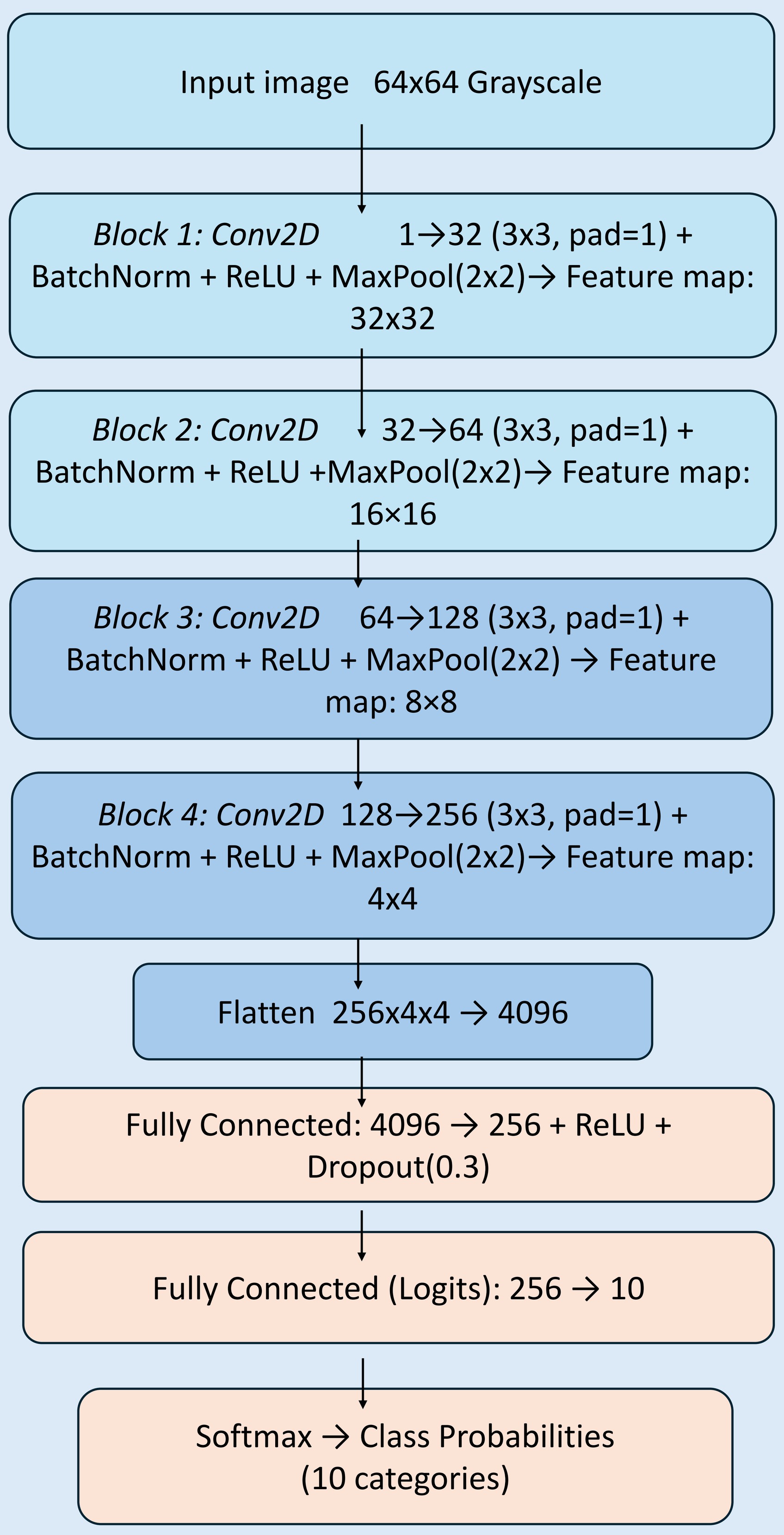


Fig. 1B. SketchCNN Architecture

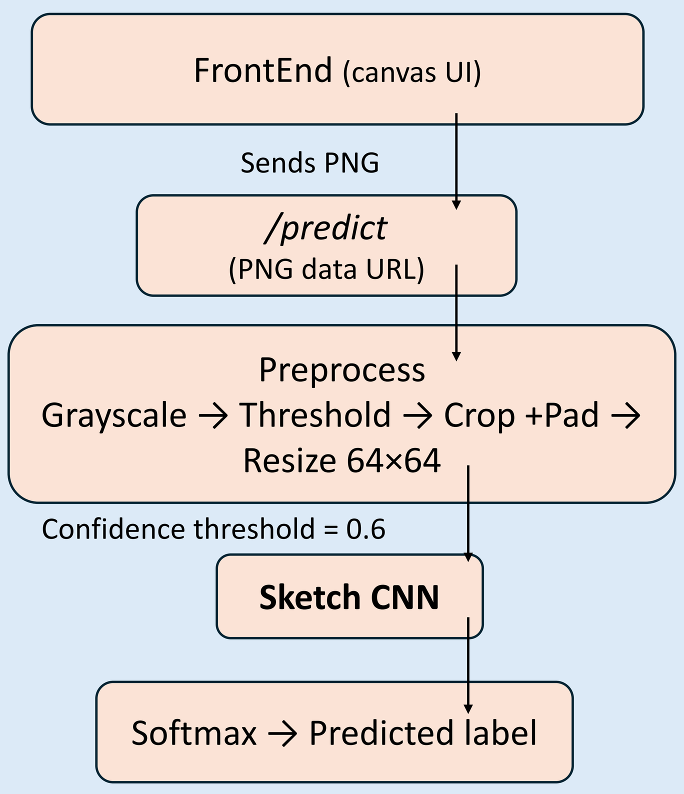


Fig. 1A. Classification Pipeline.

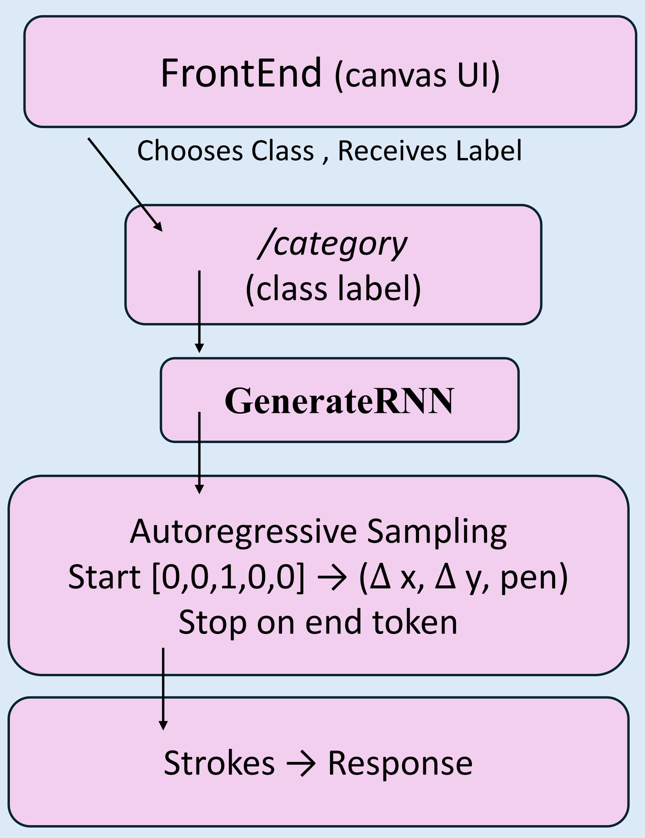


Fig. 1C. Generation Pipeline.

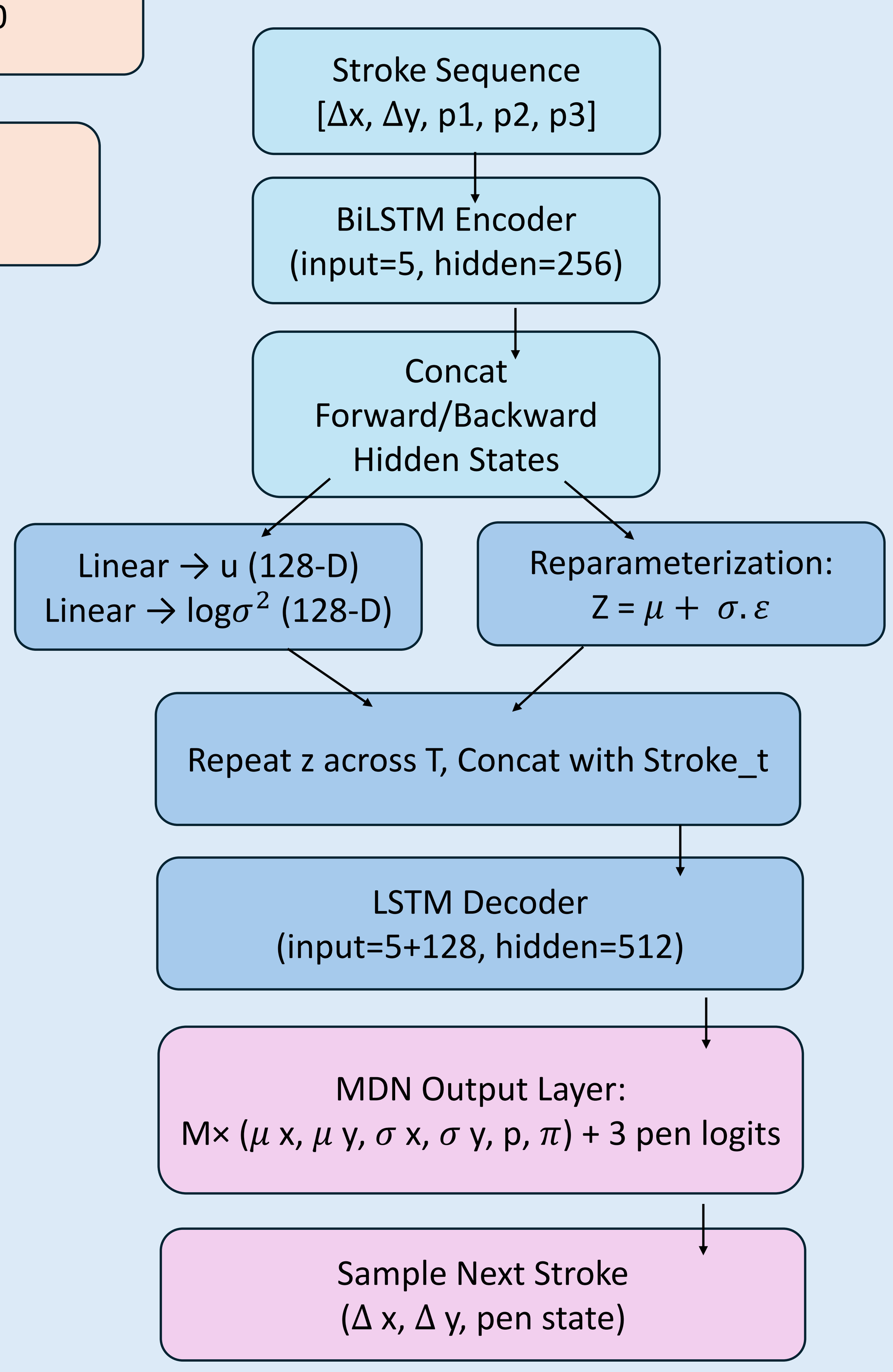


Fig. 1D. GenerateRNN Architecture

3. RESULTS

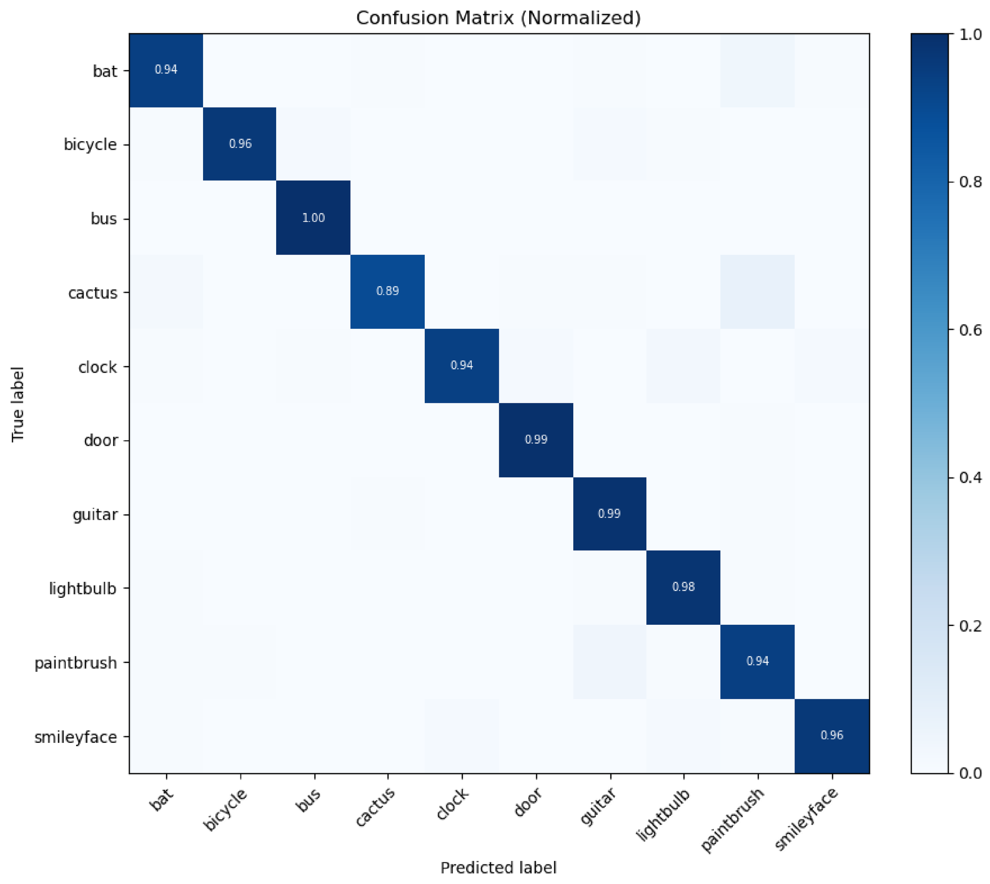
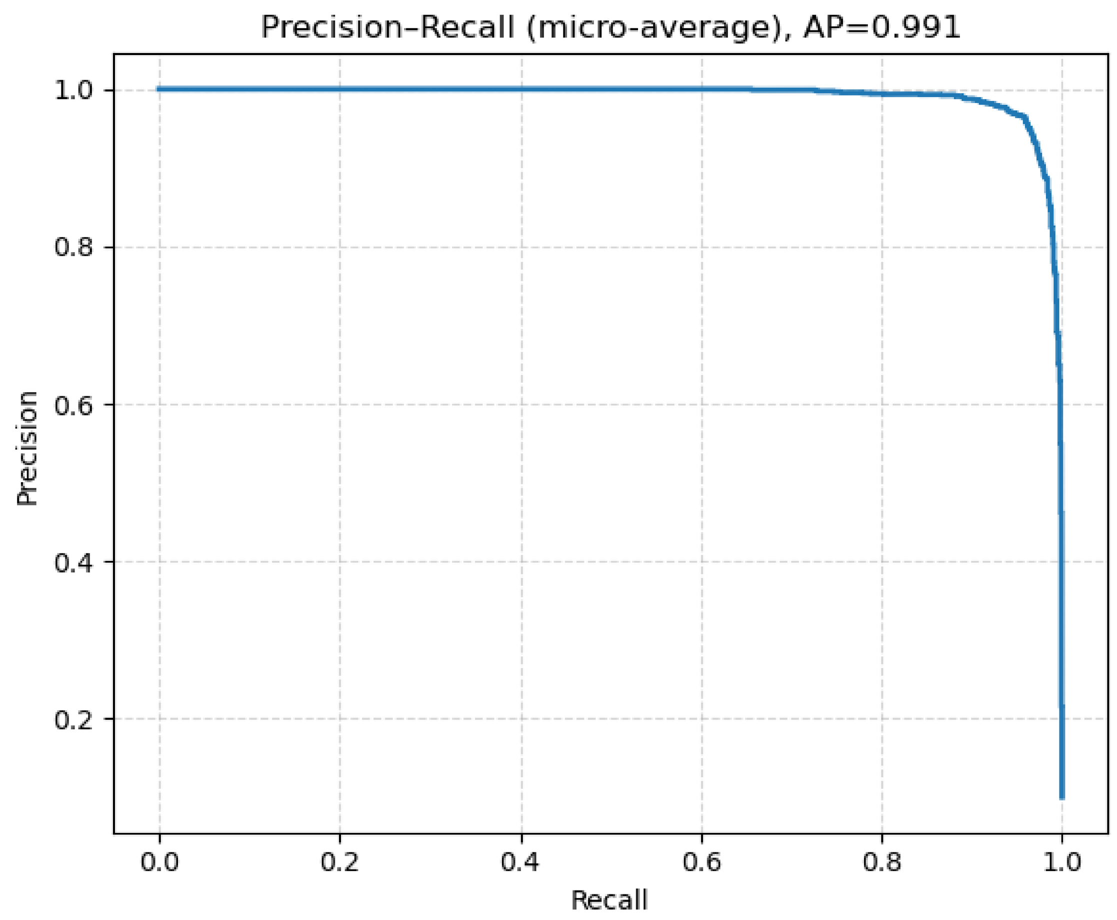


Fig. 2A. Normalized confusion matrix (test set). *Classification*

- **Overall:** strong diagonals **8/10** classes ≥ **0.94** recall.
- **Best: bus = 1.00; door ≈ 0.99**.
- **Lowest: cactus ≈ 0.89**, **paintbrush ≈ 0.94**.
- thin/elongated shapes (cactus, paintbrush) share stroke topology → occasional confusions with rounded/line-like classes.

Fig. 2B Precision–Recall *Classification*

- Curve stays near **top-right → high precision across recall**.
- **Average Precision (micro) 0.991** confirms balanced performance despite class differences.
- Thresholding: the app's **0.6 confidence** keeps precision high while rejecting uncertain cases.



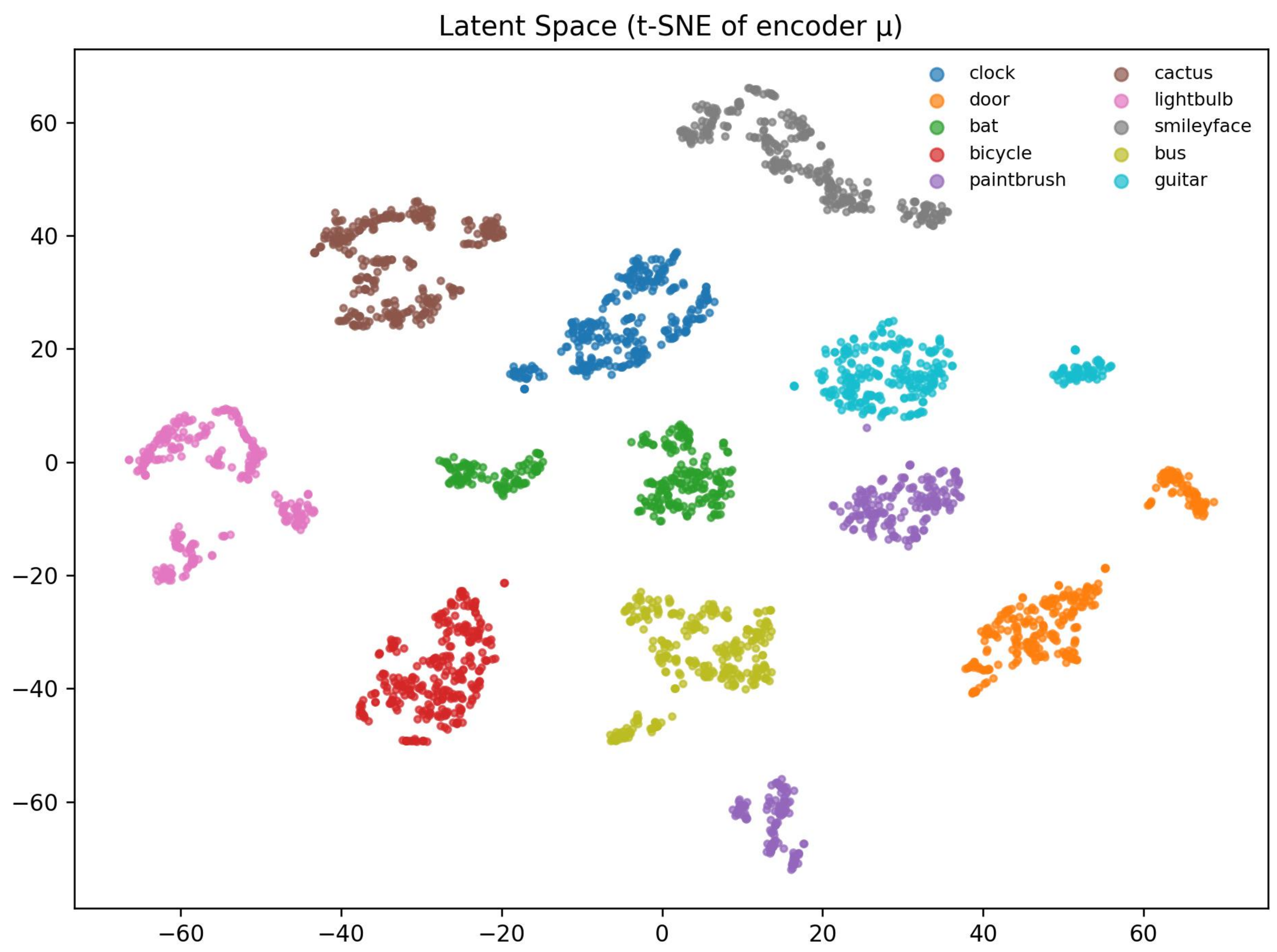


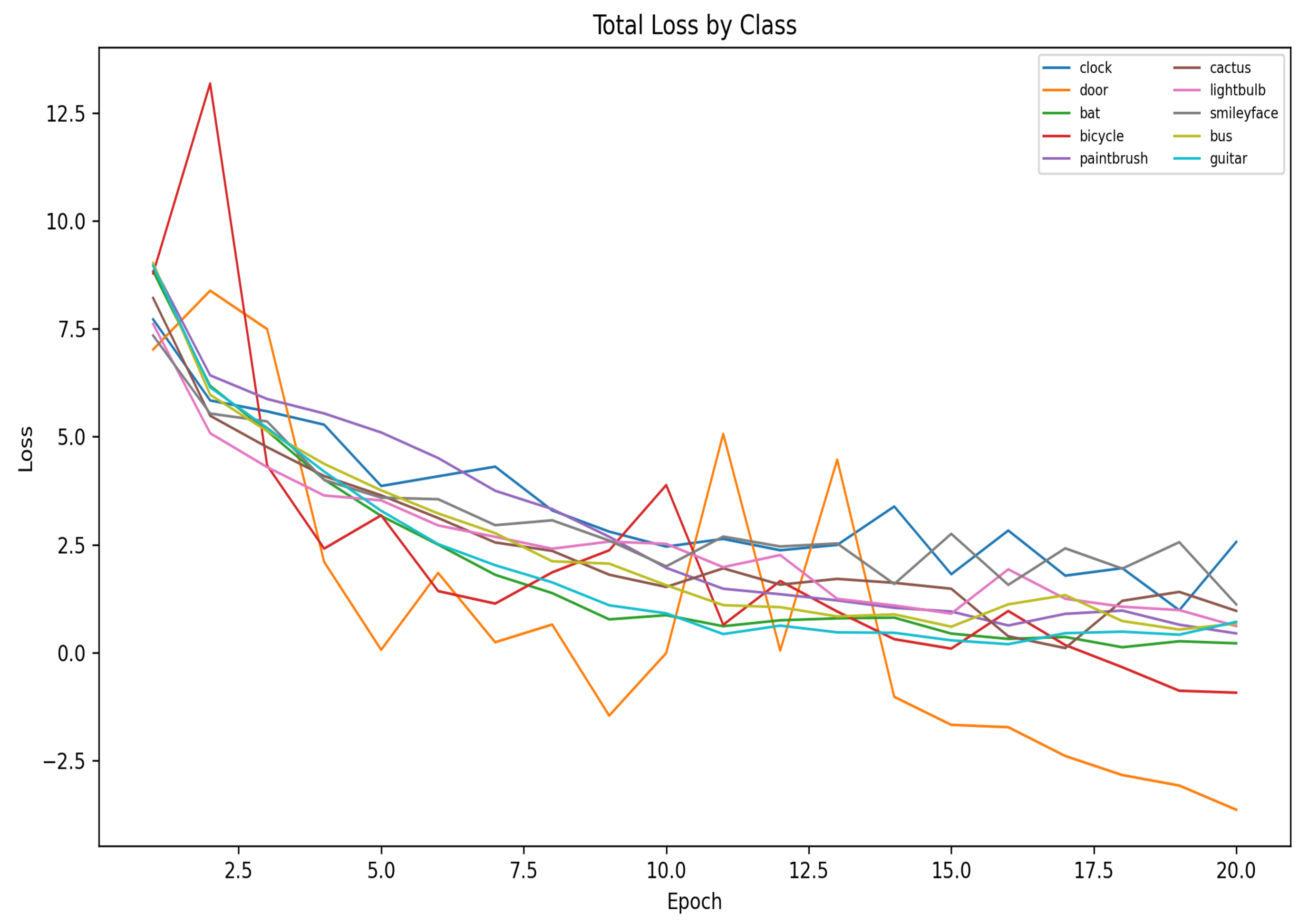
Fig. 3A Latent Space t-SNE *Generation*

- **Clear class-separable clusters** latent z captures object structure.
- **Compact clusters** indicate consistent intra-class stroke patterns.
- **Minimal overlap** → decoder can condition on z to produce class-specific strokes.

Fig. 3B Total Loss by Class *Generation*

$$\mathcal{L}_{total} = \mathcal{L}_{rec} + \beta . \mathcal{L}_{KL}$$

- **Strong downward trend** in first **5–8 epochs**; most classes **plateau ~epoch 12–15**.
- Occasional **spikes** from MDN sampling/difficulty of some classes .



4. CONCLUSION

- Our classifier achieves **96.0%** test accuracy (**Macro-F1 0.959**, **AP 0.991**); the **per-class F1** plot is uniformly high most classes ≥ 0.94 with **bus** and **door** near 0.99 and **cactus/paintbrush** slightly lower due to similar thin, line-like strokes.
- the generator learns **class-separable latents** and draws **coherent, class-conditioned** sketches showing that **compact models** can power a **reliable, interactive** tool.
- Next, we will build a **Transformer/attention-based conditional generator** and strengthen conditioning with **KL-annealing** and **probability calibration**.

Fig. 4A. Per-class F1 scores on the test set.