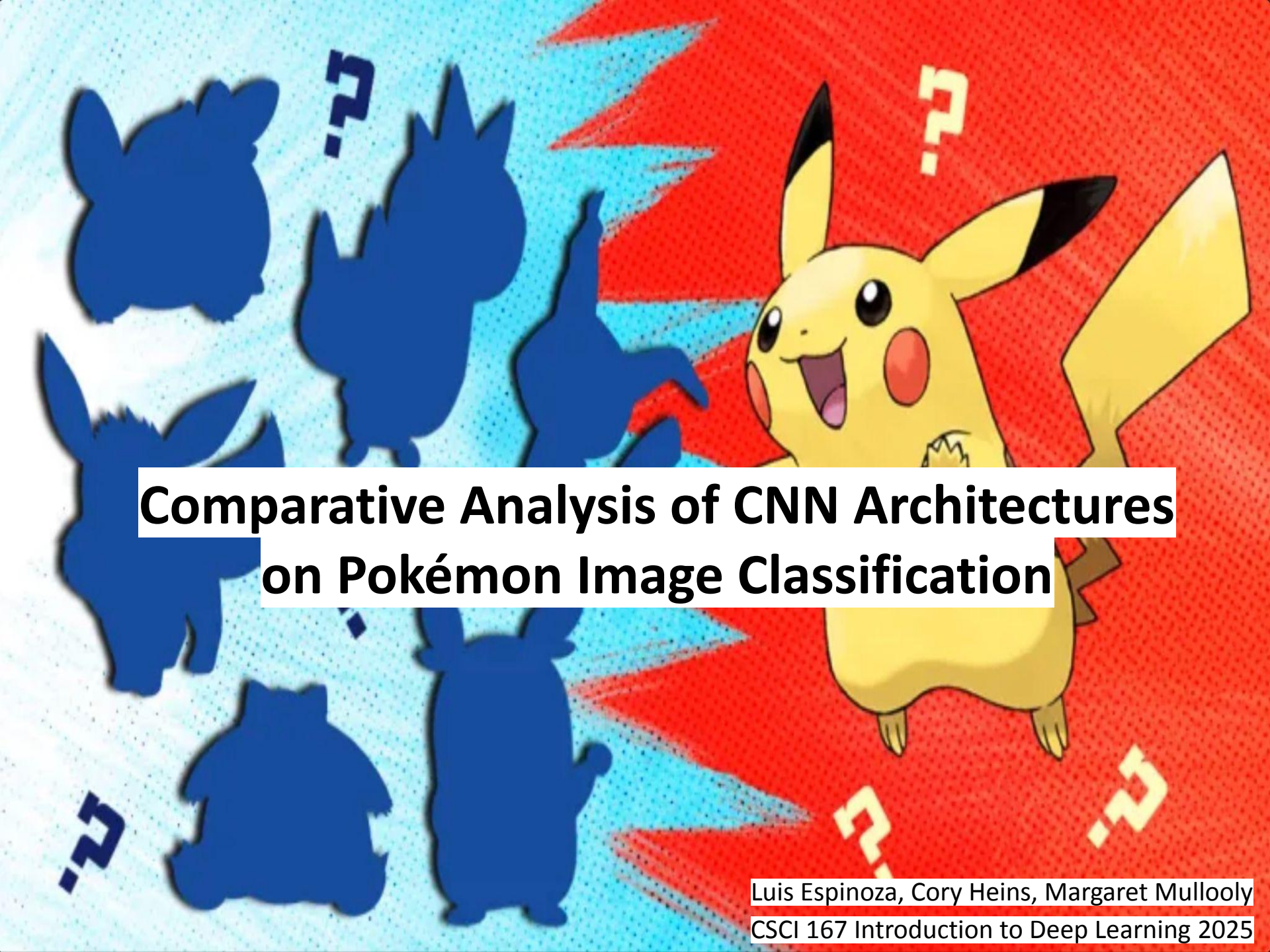
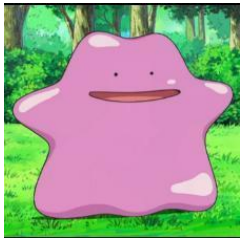


Comparative Analysis of CNN Architectures on Pokémon Image Classification

Luis Espinoza, Cory Heins, Margaret Mullooly

CSCI 167 Introduction to Deep Learning 2025

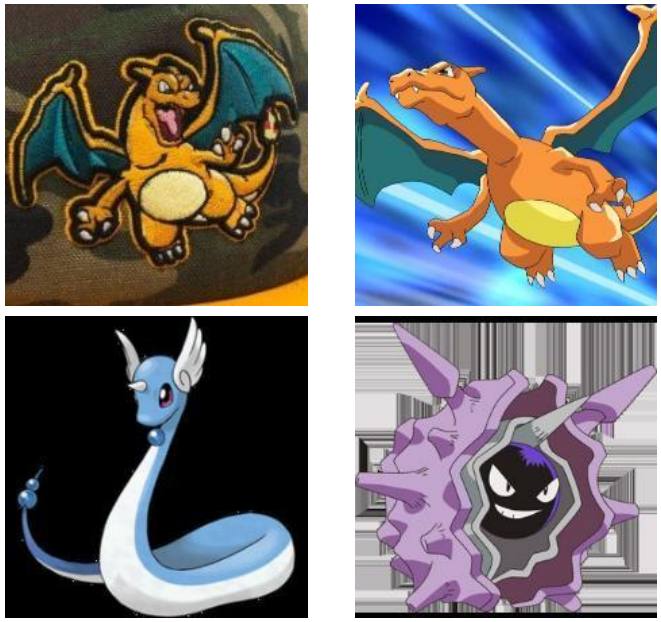
Project Overview & Goals



- Evaluate 3 convolutional neural network (CNN) architectures on a Pokémon image dataset
  - SimpleCNN, VGG11, and ResNet18
- Train models with varied hyperparameters to assess performance, convergence, and generalization.
- Goals:
  - Compare architectural impacts on learning dynamics and robustness.
  - Analyze hyperparameter effects on optimization and stability.
  - Derive insights into transferable design choices for image classification tasks.

Dataset Overview

- Dataset: Pokémon images split into train, validation, and test sets.
  - Batch sizes: 128 train, 256 val, 256 test
- 110 classes representing distinct Pokémon species (e.g., Porygon, Goldeen, Hitmonlee, ..., Raichu).
- Image specifications: 64×64 resolution, RGB format



- Preprocessing:
  - Training: random resized crop, horizontal flip, normalization.
  - Validation/Test: resize, center crop, normalization.
  - Rationale: Diverse visual features (shapes, colors, poses) support fine-grained classification analysis.

Architecture 1: SimpleCNN

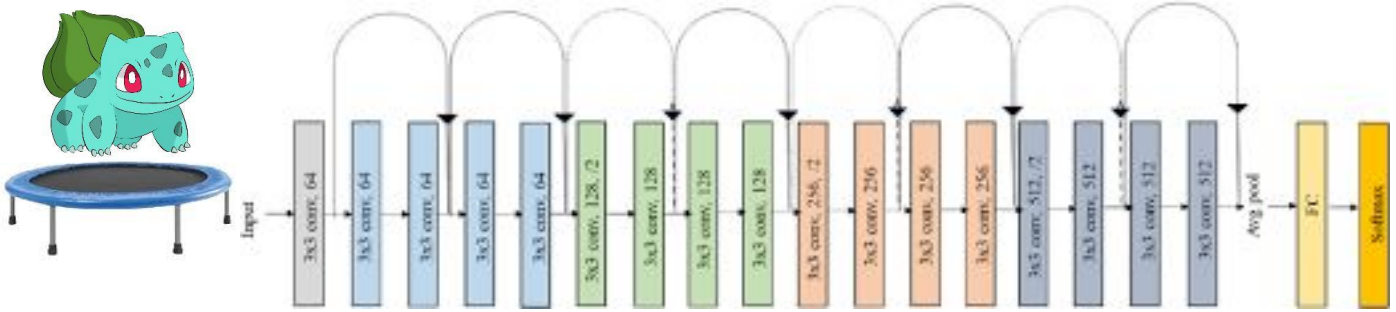
- Lightweight baseline CNN with stacked 3×3 convolutional layers, batch normalization, ReLU activations, and max pooling.
  - 3 conv blocks (64→128→256), each with Conv–BN–ReLU–Conv–BN–ReLU–MaxPool
- Fully connected classifier with dropout, configured for 110-class output.
  - 3 conv blocks (64→128→256), each with Conv–BN–ReLU–Conv–BN–ReLU–MaxPool
- Focuses on basic feature extraction without advanced mechanisms.
- Why we chose this: Serves as a control to highlight limitations of shallow networks on complex Pokémon images.

Architecture 2: VGG11

- Deeper network using uniform 3×3 convolutions, batch normalization, ReLU, and max pooling.
  - Two fully connected layers with dropout, ending in a 110-class classifier.
  - Dense head (512×4×4 → 512 → 512 → 110 with dropout)
- Emphasizes depth through repetitive small-kernel stacking.
- Why we chose this: Demonstrates benefits and challenges of deeper plain CNNs (e.g., vanishing gradients).

Architecture 3: ResNet18

- Residual network with 18 layers and shortcut connections for identity mappings.
  - 4 stages (64, 128, 256, 512) + global average pooling → 110-way final linear layer
  - Improved gradient flow via residual blocks with convolutions and batch normalization.
- Why we chose this: Illustrates advantages of residual learning for deeper training and better generalization.



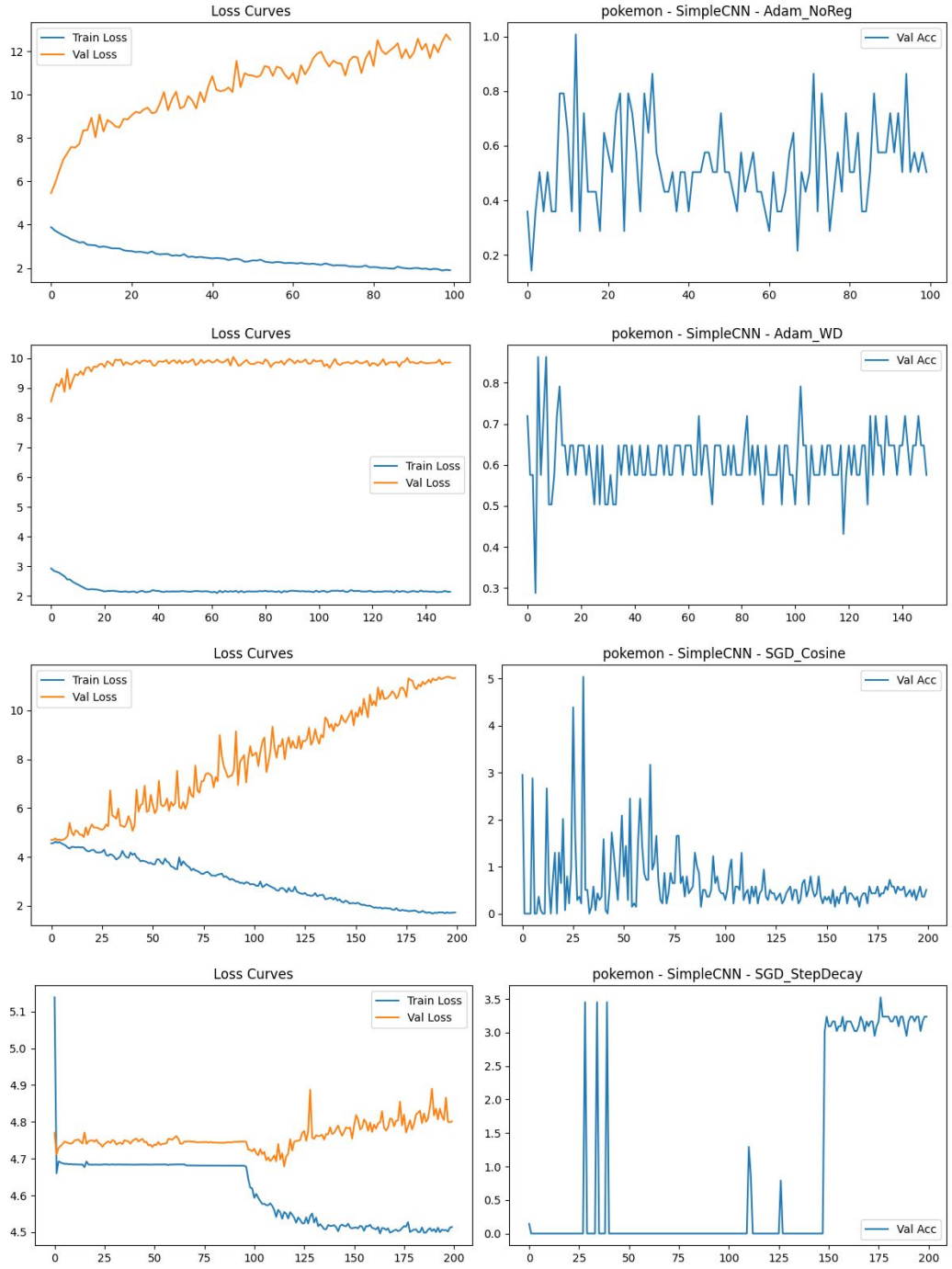
Experimental Setup

- Models: SimpleCNN, VGG11, ResNet18
- Hyperparameter configurations:

Config Name	Optimizer	LR	Momentum	Weight Decay	Scheduler	Epochs
SGD_StepDecay	SGD	0.1	0.9	5e-4	StepLR	200
SGD_Cosine	SGD	0.1	0.9	5e-4	CosineAnnealingLR	200
Adam_NoReg	Adam	0.001	N/A	0.0	None	100
Adam_WD	Adam	0.0005	N/A	1e-4	ReduceLROnPlateau	150

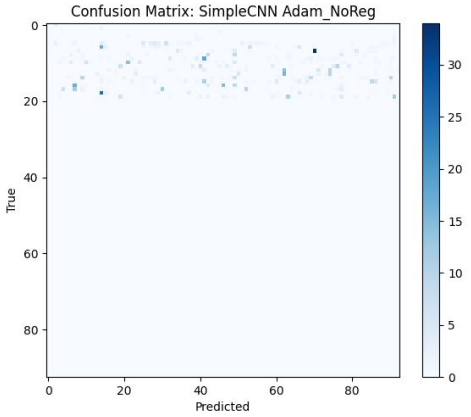
Learning Curves SimpleCNN

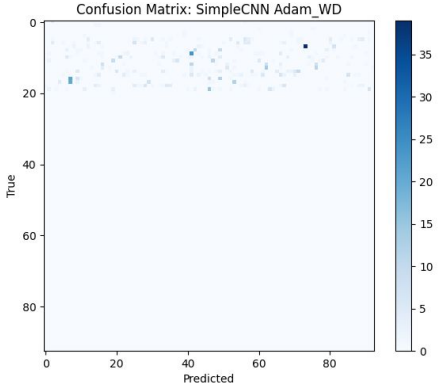
- Training loss steadily decreases
- Validation loss consistently increases
- Validation accuracy remains low and highly unstable
- SGD (StepDecay / CosineAnnealing) shows slightly more stable curves but no significant improvement in accuracy
- Adam yields faster training loss reduction but no generalization gains

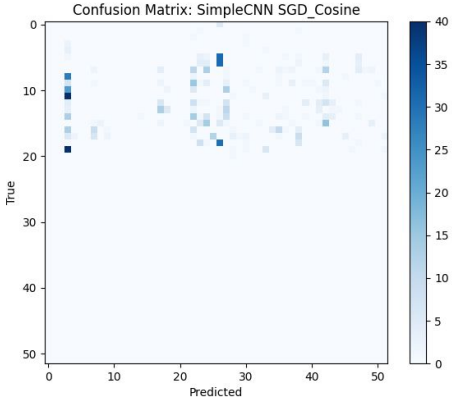


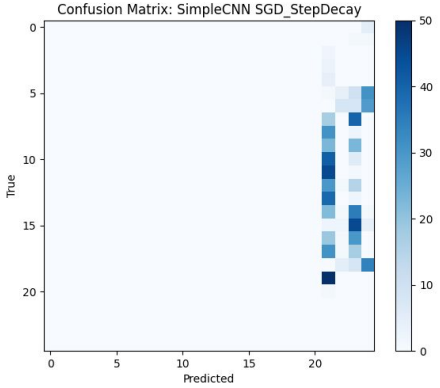
Simple CNN - Confusion Matrix

- Some configurations (SGD_StepDecay, SGD_Cosine) exhibit mode collapse, repeatedly predicting only a handful of classes
- Adam_NoReg and Adam_WD scatter predictions across many wrong categories, essentially uniform guessing.



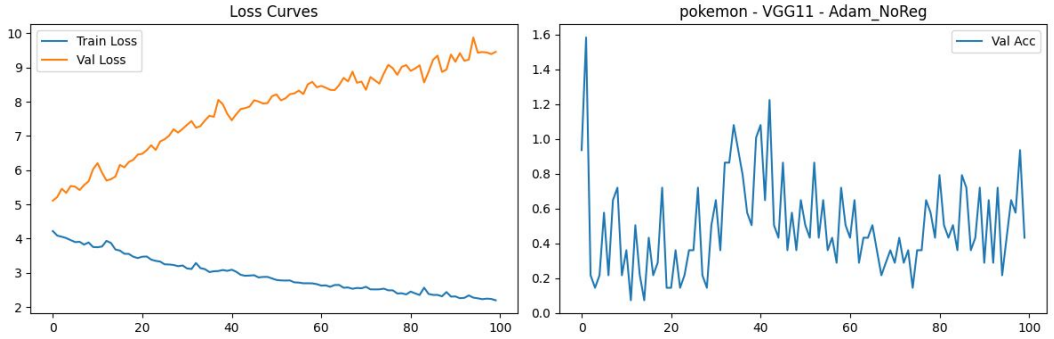


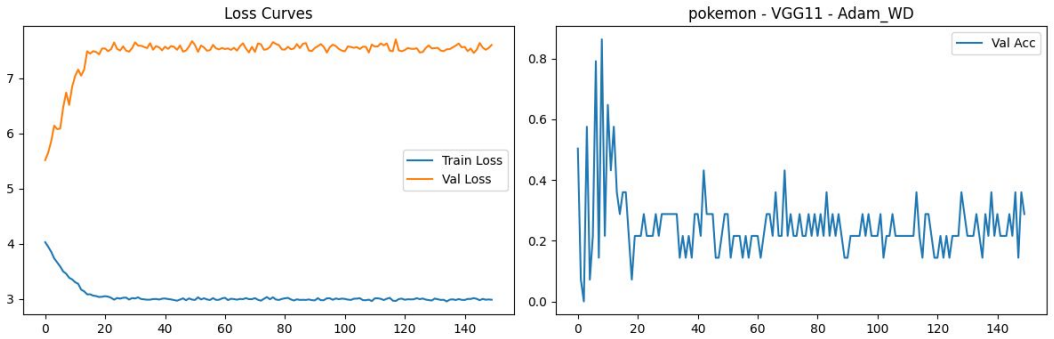


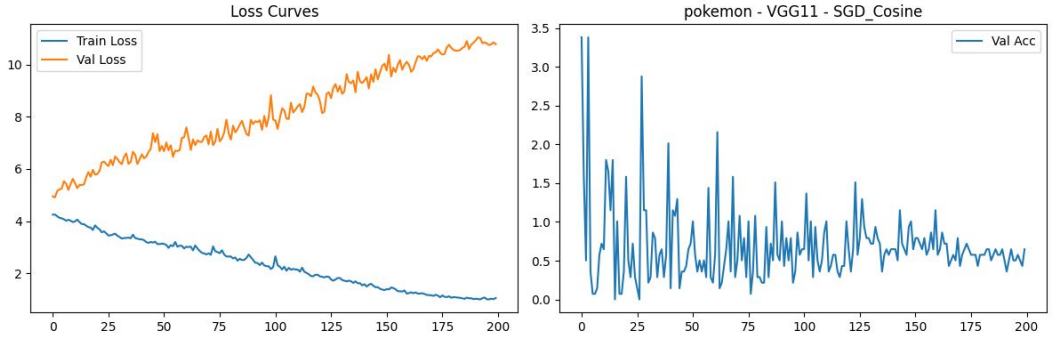


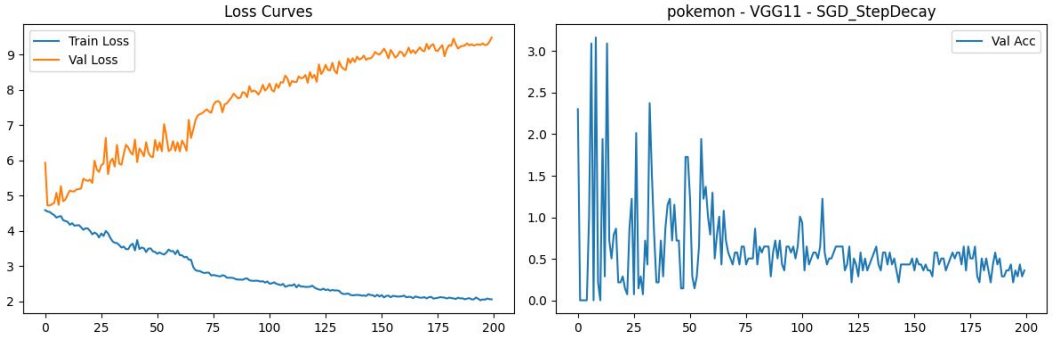
Learning Curves VGG11

- All VGG11 learning curves show:
  - Train loss dropping steadily
  - Validation loss rising over time
  - Validation accuracy fluctuating near random
  - This pattern shows the model memorizes training images but fails to learn class-discriminative features that transfer to validation images.



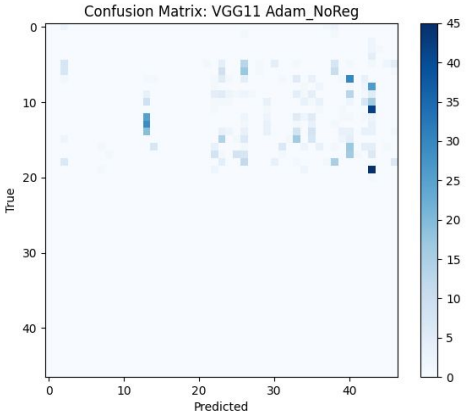


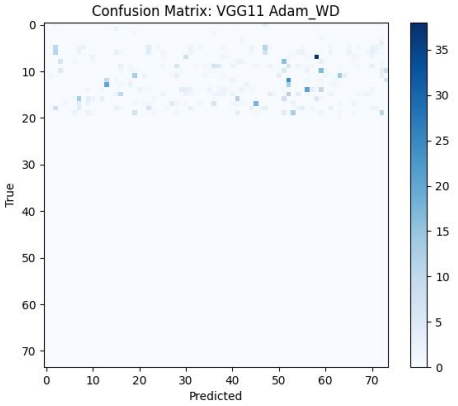


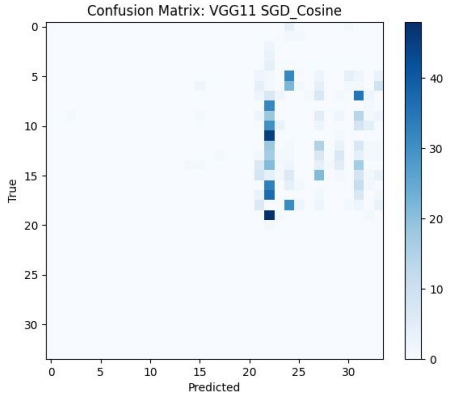


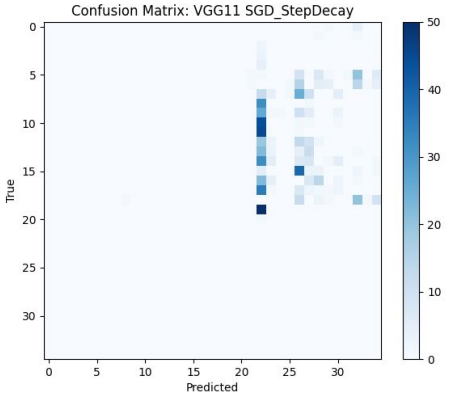
VGG11 - Confusion Matrix

- Consistently fails to generalize on this dataset
  - Across all hyperparameter configurations validation accuracy stays between 0–6%
  - Most predictions fall into only 4–10 output classes (out of 110).
  - Adam configs scatter predictions across many incorrect classes (uniform guessing)



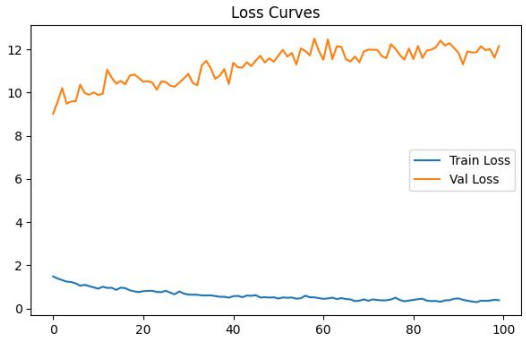


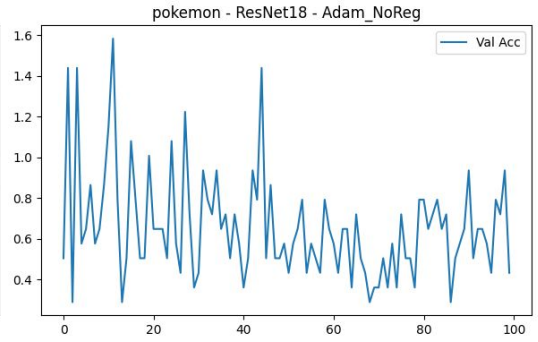


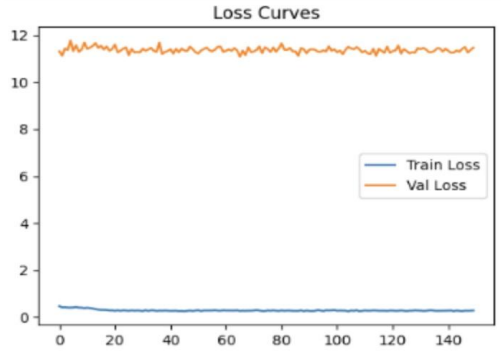


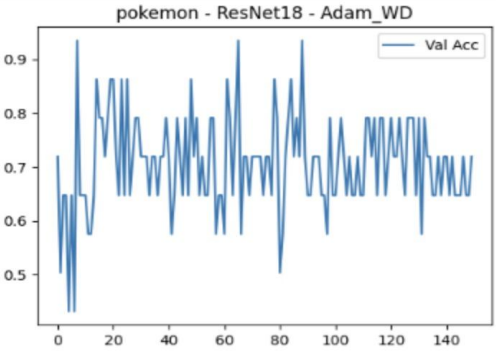
Learning Curves ResNet18

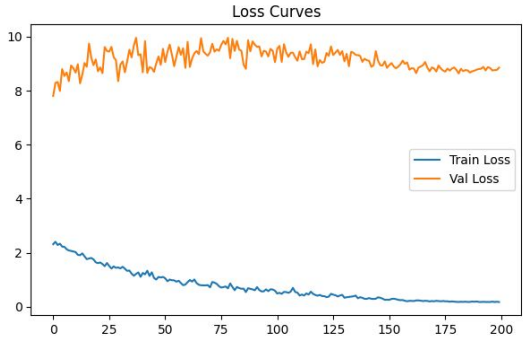
- Training loss decreases smoothly across all optimizers, indicating successful optimization and effective use of residual connections.
  - Validation accuracy is unstable and low (0.5–2.8%), but consistently higher than SimpleCNN and VGG11, meaning ResNet18 extracts more useful features.
  - SGD produces the highest test accuracies (1–2.2%) but still fluctuates heavily.
  - Adam configs train fastest but generalize poorly, especially without weight decay.
- ResNet18 learns more robust representations than the other two models, but still struggles with the 110-class Pokémon dataset.

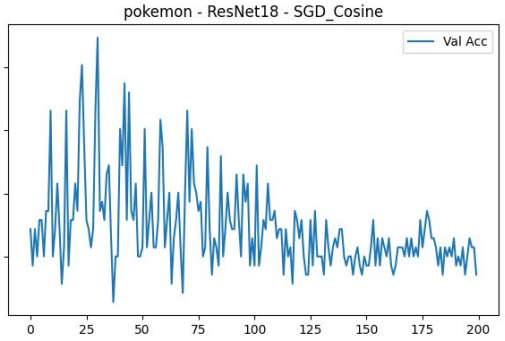


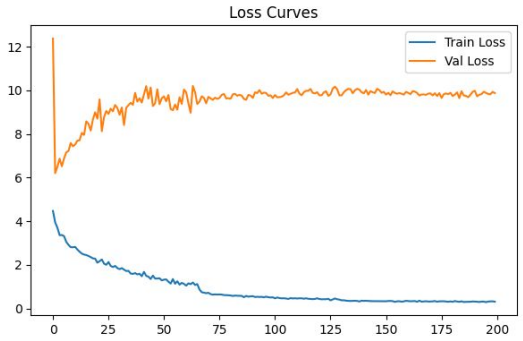


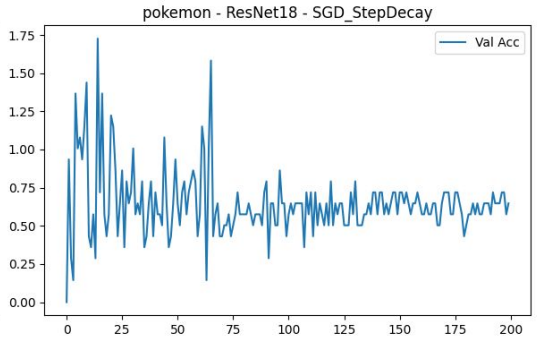






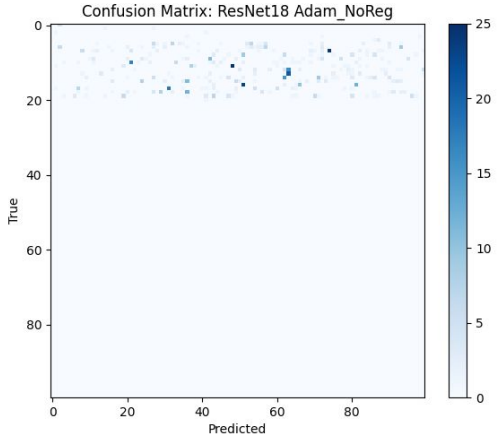


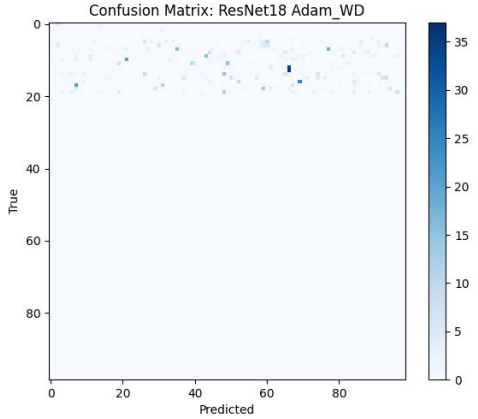


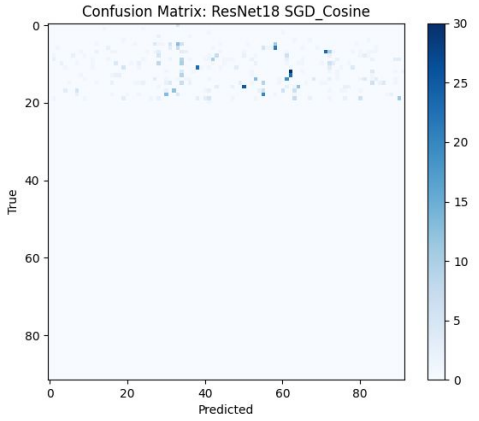


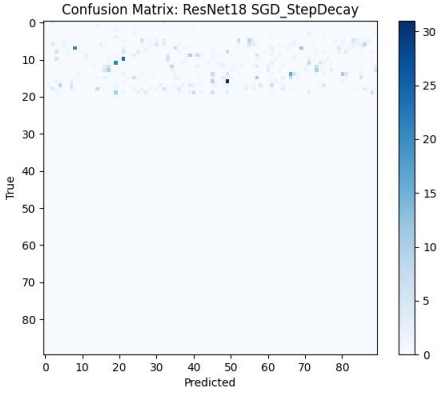
ResNet18 Confusion Matrix

- Many predictions concentrate in a band across certain classes, but the model fails to consistently match true labels.
- Compared to SimpleCNN/VGG11, ResNet18 still cannot separate 110 visually similar classes on this limited dataset.

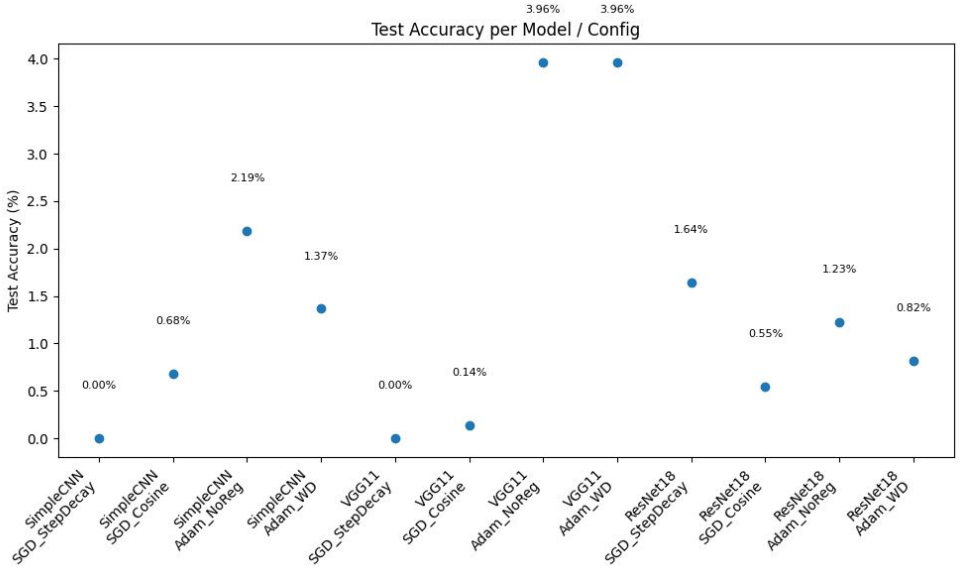








Key Findings



- All three models struggled significantly on this fine-grained 110-class Pokémon dataset when trained from scratch.

- Optimizer & scheduler choice mattered, with Adam variants producing higher score for SimpleCNN and VGG11, and SGD variants producing highest test scores for ResNet18.
- Confusion matrices lacked diagonal structure, indicating that models weren't learning meaningful class-discriminative features.

Cross-Dataset Insights

- Architectures that work well on ImageNet do not transfer well when trained from scratch on small datasets.
- Hyperparameters that succeed on CIFAR/ImageNet (SGD + cosine decay) did not translate into meaningful gains here.
- Results imply that transfer learning or pretraining is essential for small-scale, high-class image tasks.

What Surprised Me

- Even deep models failed completely. We expected ResNet18 to achieve at least 20-30% accuracy, but it never learned meaningful decision boundaries.
- SGD sometimes performed better than Adam, despite slower initial convergence, showing that “fast loss decrease” doesn’t mean better generalization.
- Confusion matrices were essentially random, with no diagonal structure
- The dataset’s diversity + class imbalance had a much larger effect than anticipated.

Limitations & Future Work

- Limitations:
  - Dataset small relative to number of classes; some classes had very few samples.
    - Training set had 110 classes, but the test set had (unannounced) 150 classes.
  - Hyperparameter systematic tuning of LR, WD, and batch size.

- Future Work:
  - Apply pretrained backbones and fine-tune; likely dramatic improvement.
  - Try class-balanced sampling to stabilize training.

Thanks for listening!

- Any Questions?

- Pokemon dataset → https://huggingface.co/datasets/keremberke/pokemon-classification