

CLASSIFYING N64 IMAGE TEXTURES WITH CONVOLUTIONAL NEURAL NETWORKS.



Max Anderson, Galvanize Data Science Immersive, 2021.

PROJECT DESCRIPTION

- Attempt to use **Convolutional Neural Networks** in order to classify image textures from **Nintendo 64 games** by use in game development or game modding.
- Images are taken from popular Mario modding resource **smwcentral.net**
- Data was **hand labeled** into what I consider to be the most useful categories based on personal experience
- Three **CNN models** were successfully trained and compared based on accuracy



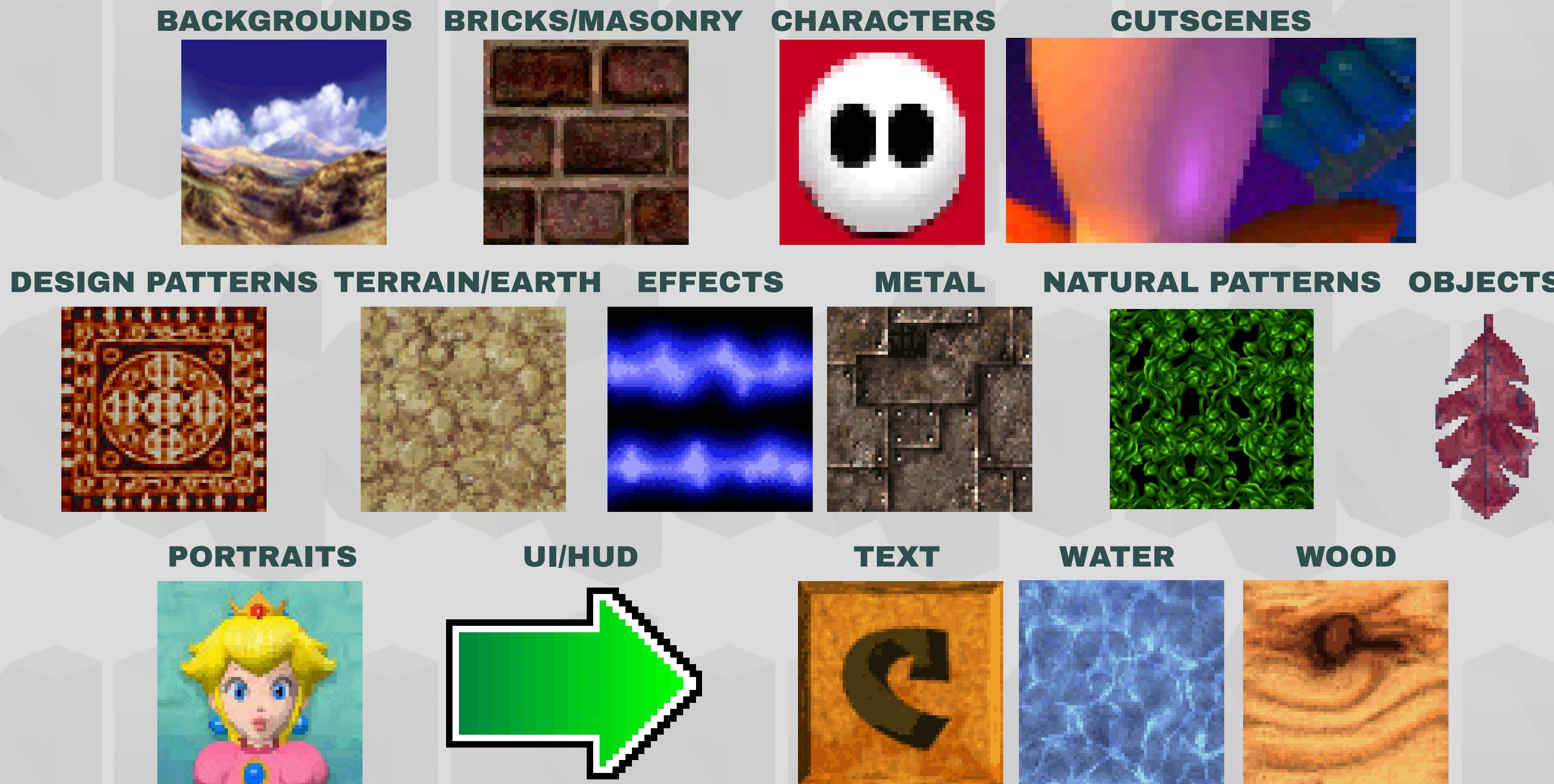
PROJECT SCOPE

- **Pre-capstone:** Finding data, data cleaning
- **Day One:** Labeling, image generation, baseline model prelim
- **Day Two:** Labeling, baseline model work, building other models, callbacks
- **Day Three:** Tuning hyperparameters, refining callbacks, Google CoLab setup
- **Day Four:** Generating graphs and visuals, presentation work
- **Day Five:** Presentation refinement and rehearsal, Github



DATA ORGANIZATION

- Around 17,000 .png and .bmp images were taken from smwcentral.net
- Most images were 32x32 pixels in RGBA colorspace
- Around 5,000 images were successfully labeled into 15 categories:



DATA ISSUES

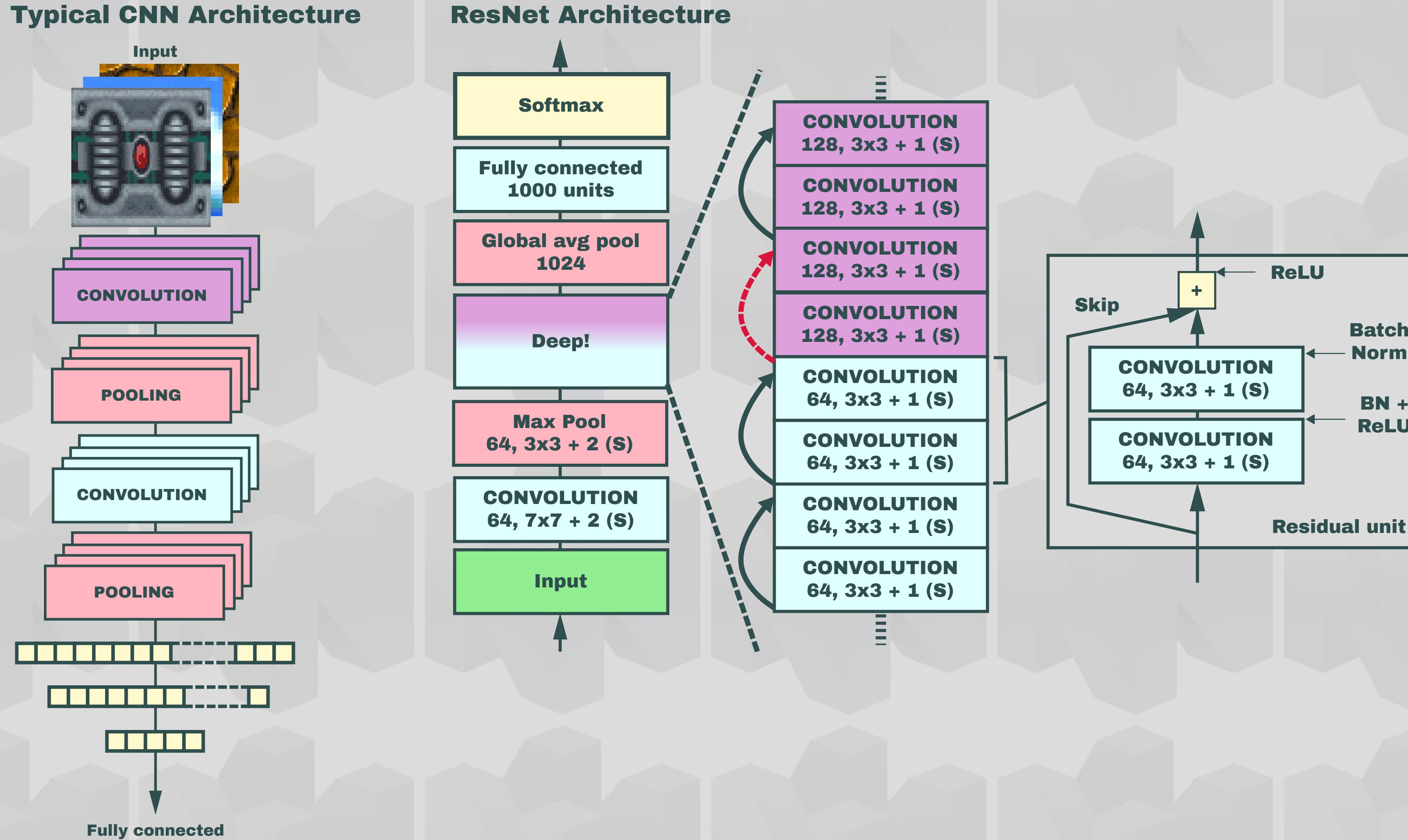
- Hand labeling is time intensive and inconsistent
- Unbalanced labels
- Differences in artistic style between games
- Differences in texture utility between games
- Low resolution prevented the use of some CNN models (Inception)
- Small set of images limits amount of epochs, steps per epochs and validation steps

MODEL SUMMARIES

- Used **simple sequential model** with 2 convolutional layers as baseline
- Used **deeper sequential model** with more layers as intermediate
- Used **ResNet 34** model as best performing model
- Attempted an **Inception** model, but did not work with the low resolution RGBA images

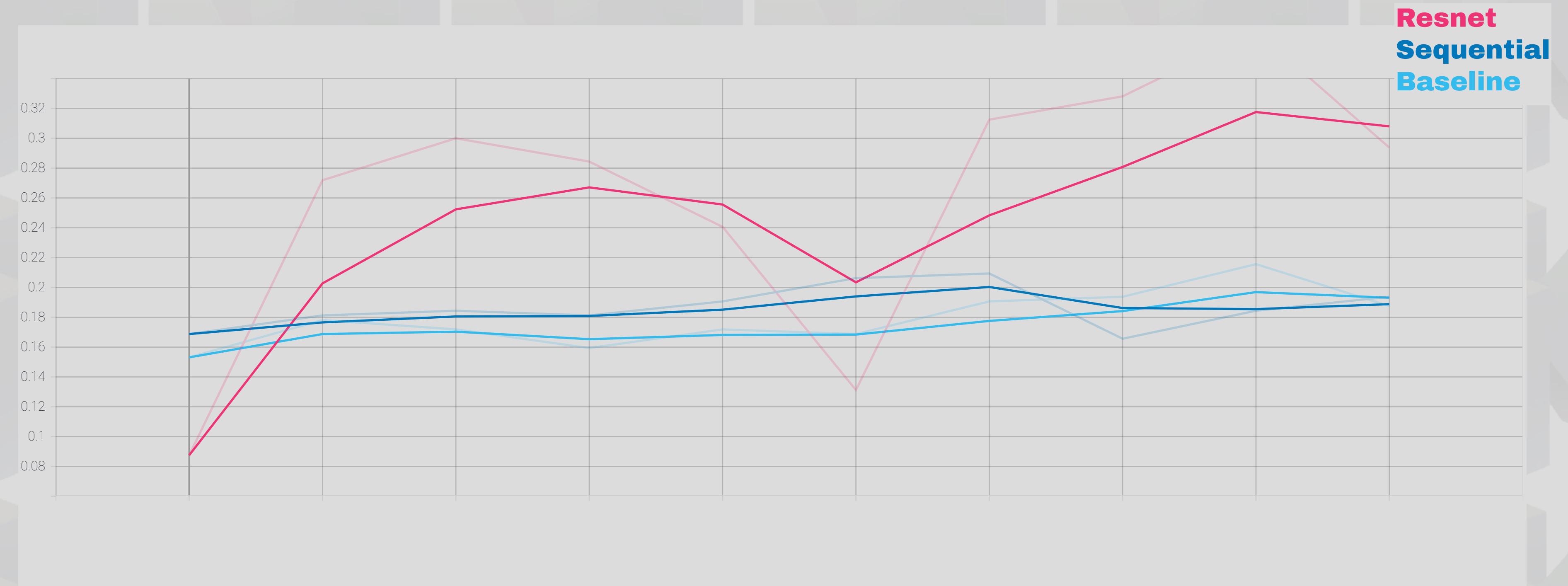


MODEL SUMMARIES



Diagrams taken from *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, 2nd Edition* by Aurélien Géron

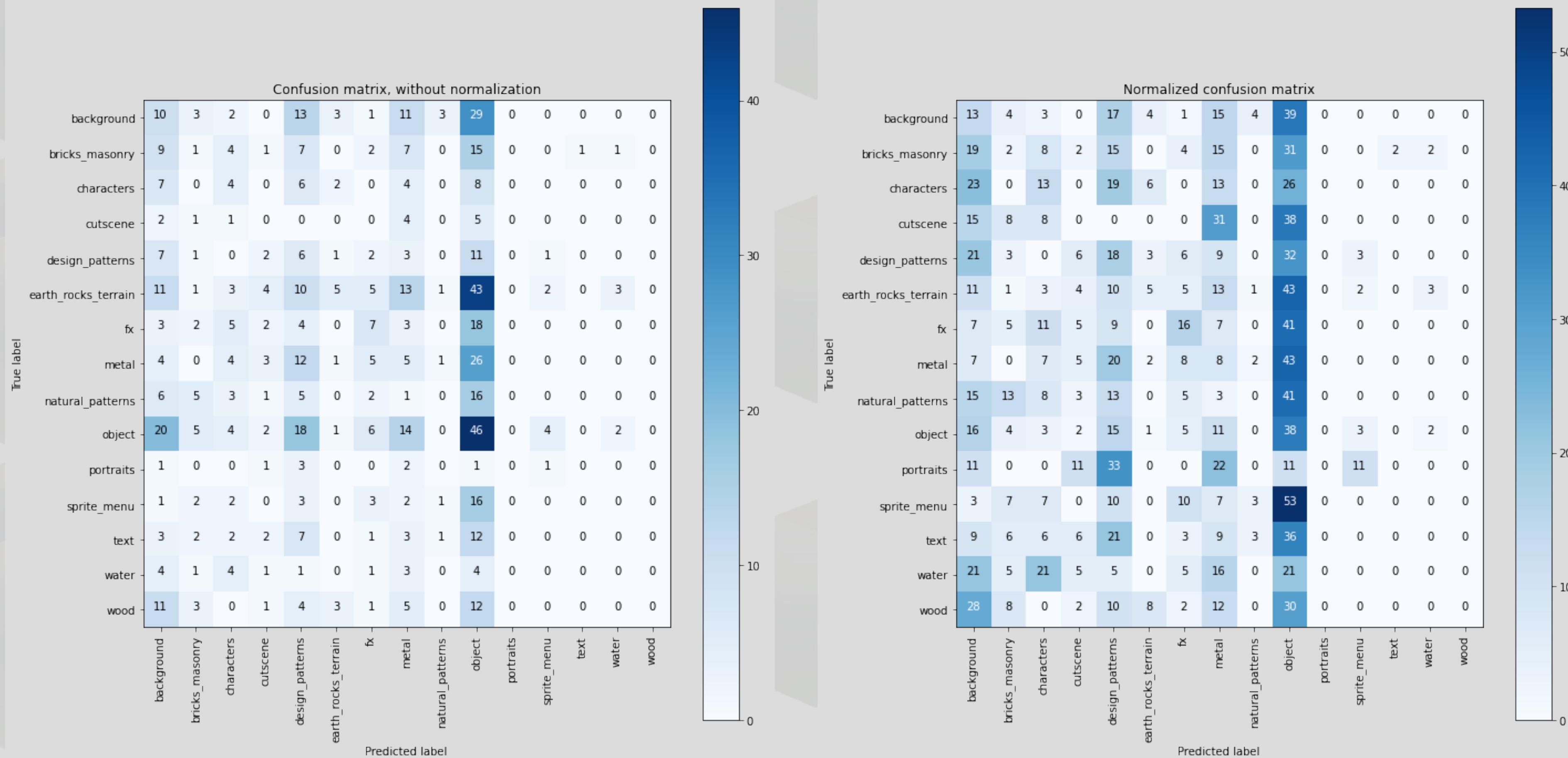
ACCURACY SCORES



- 80 epochs
- 10 epoch steps
- 10 validation steps

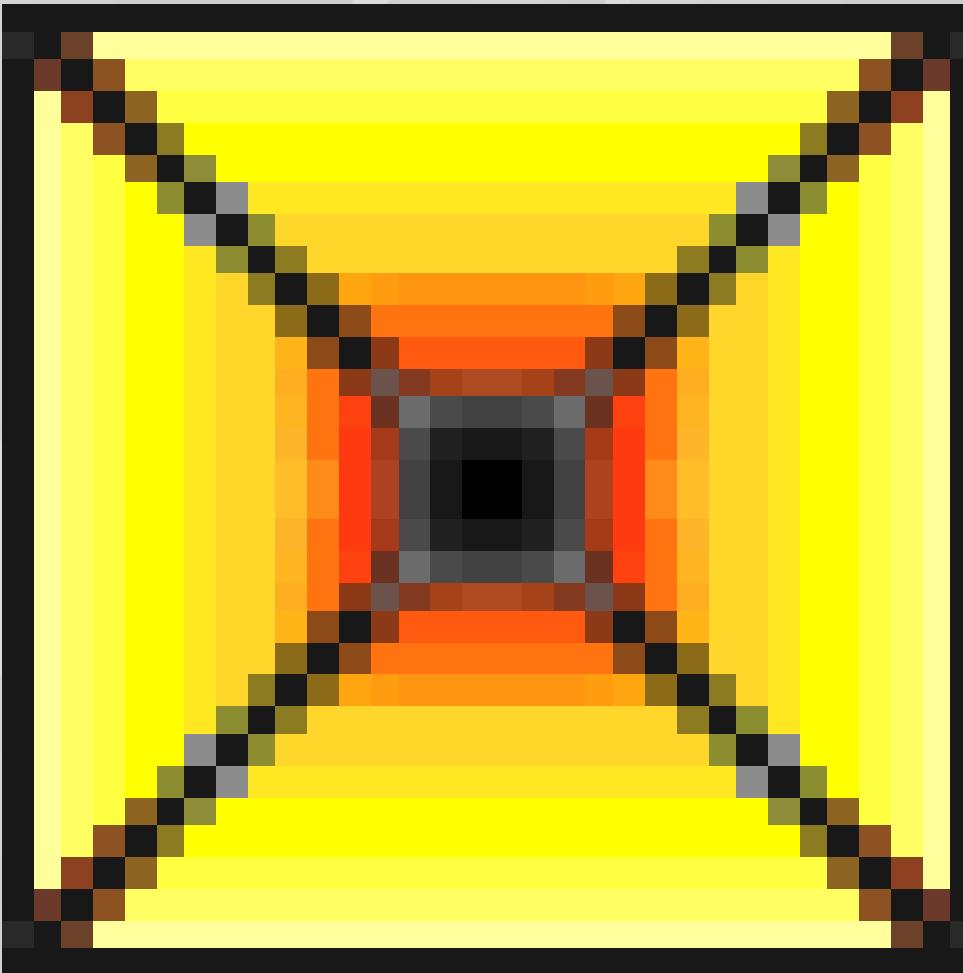
- Baseline accuracy: 0.19
- Sequential accuracy: 0.2
- ResNet accuracy: 0.32

RESNET CONFUSION MATRICES

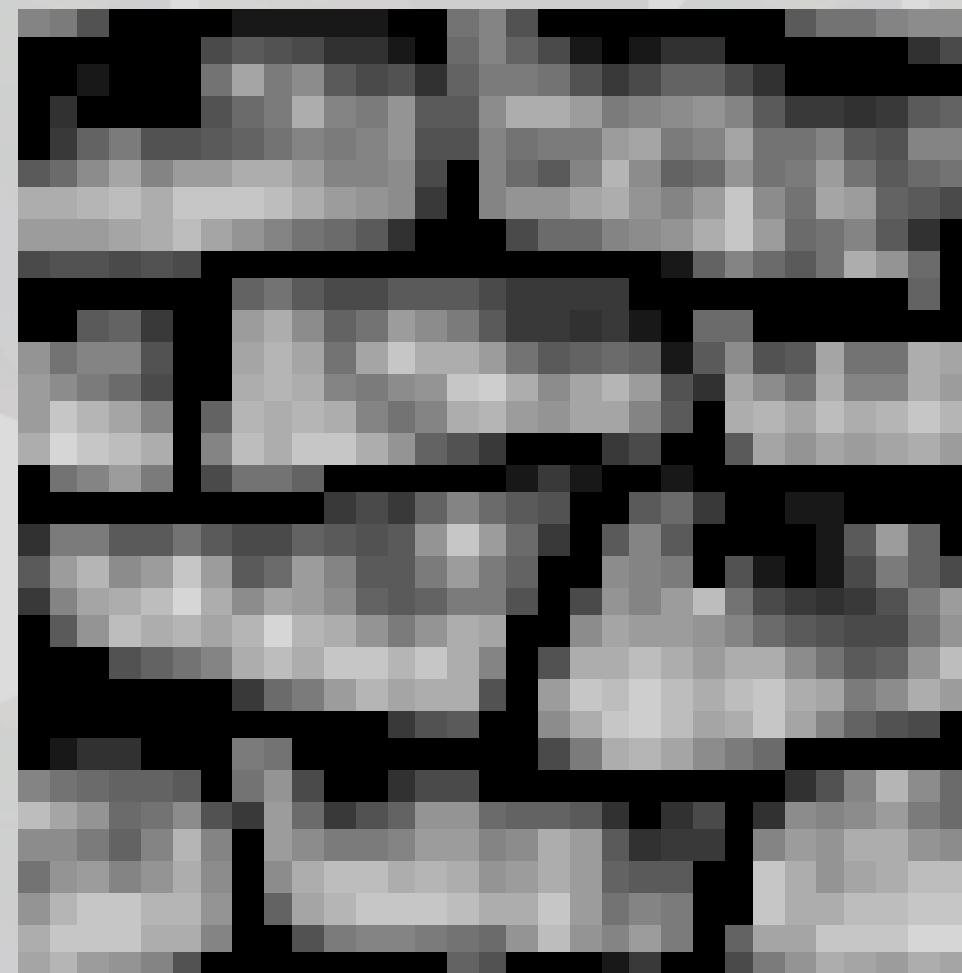


PREDICTION SAMPLES

Predicted: object
Actual: object
Confidence:0.6



Predicted: bricks_masonry
Actual: bricks_masonry
Confidence:0.17



Predicted: object
Actual: metal
Confidence:0.6



Predicted: bricks_masonry
Actual: object
Confidence:0.17



NEXT STEPS

- Find a comparable CCO or open source set of image textures
- Reassess categories
- Try more advanced models (pretrained models)
- Generate new image textures from trained models

