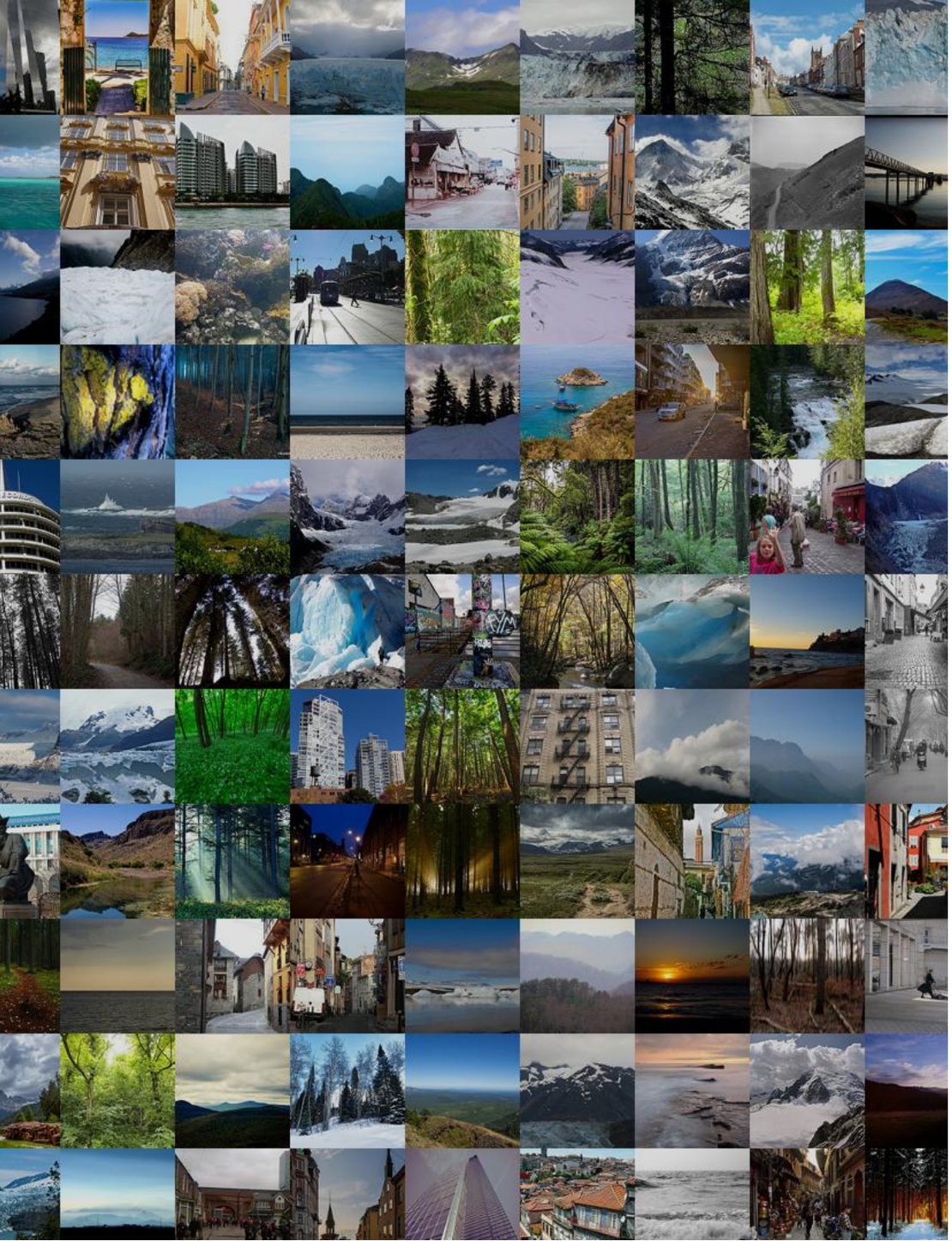
Image Archiving Model

Date:
06/27/2023

Presented by:
Ali Hijazy



Objective

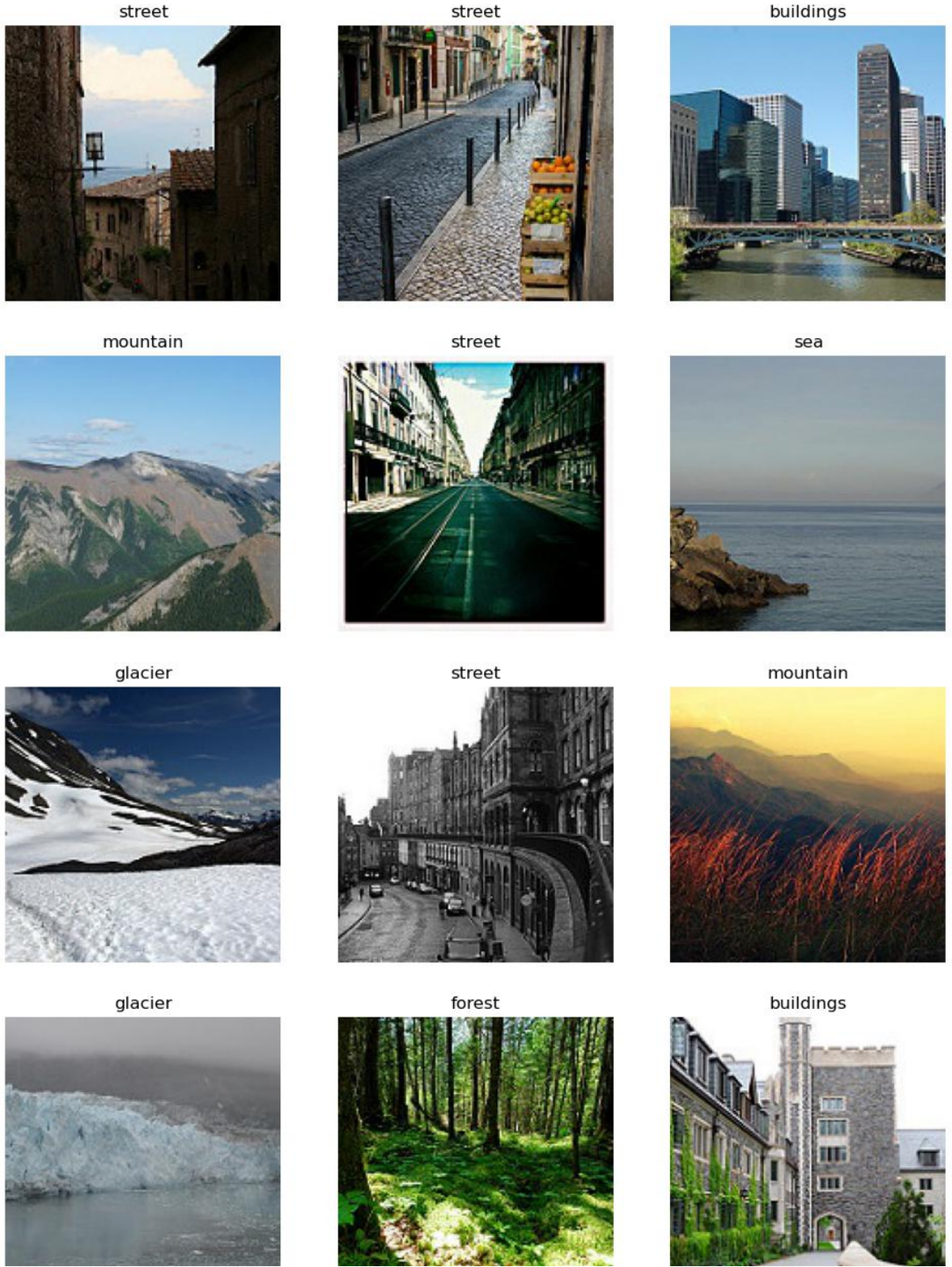
Create a model capable of distinguishing between six distinct types of scenes.

Use

- Streamlines the image archiving workflow by automating the classification task.

Operational Benefits

- Significantly reduces the time required for image classification during the archiving process
- Enables better searchability and retrieval of archived images based on scene type.



DATASET

kaggle

The dataset is 17000 labeled images divided into 6 classes

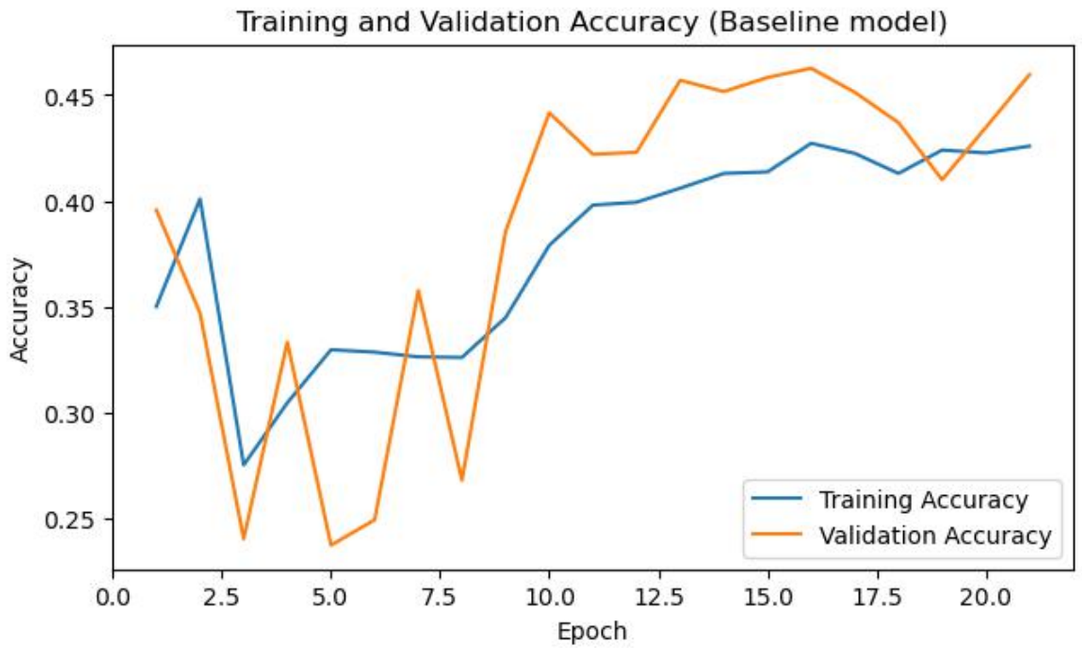
images are colored 150 * 150 pixels

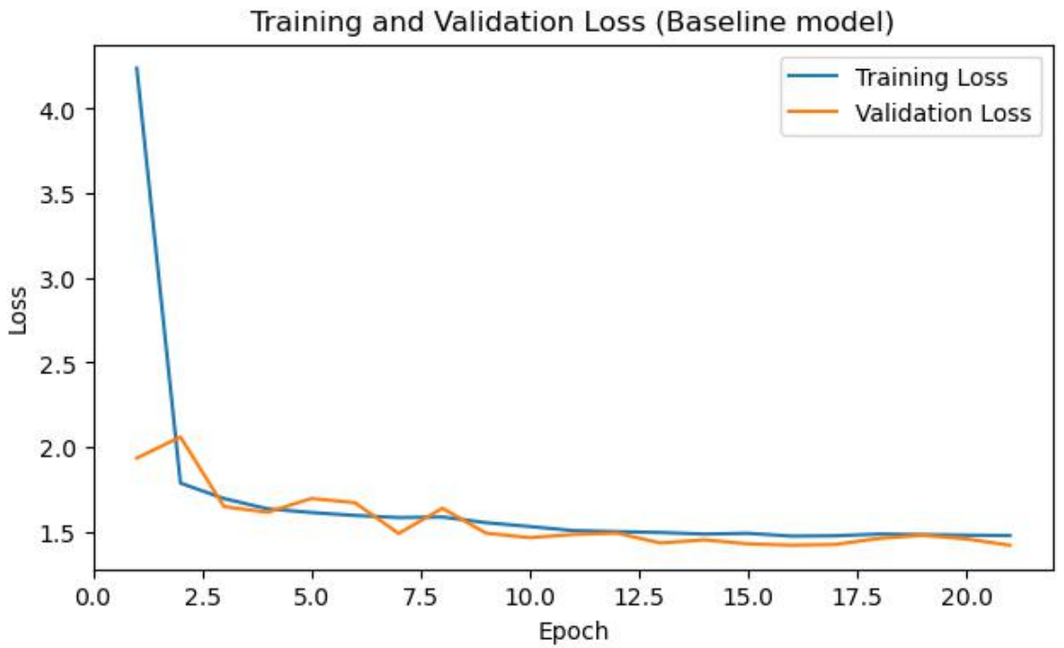
Target classes

- **Buildings**
  Train Data: 2191
  Test Data: 437
- **Forest**
  Train Data: 2271
  Test Data: 474
- **Glacier**
  Train Data: 2404
  Test Data: 553
- **Mountain**
  Train Data: 2512
  Test Data: 525
- **Sea**
  Train Data: 2274
  Test Data: 510
- **Street**
  Train Data: 2382
  Test Data: 501

Baseline Model (Neural network)

Test: 0.46
Train: 0.43

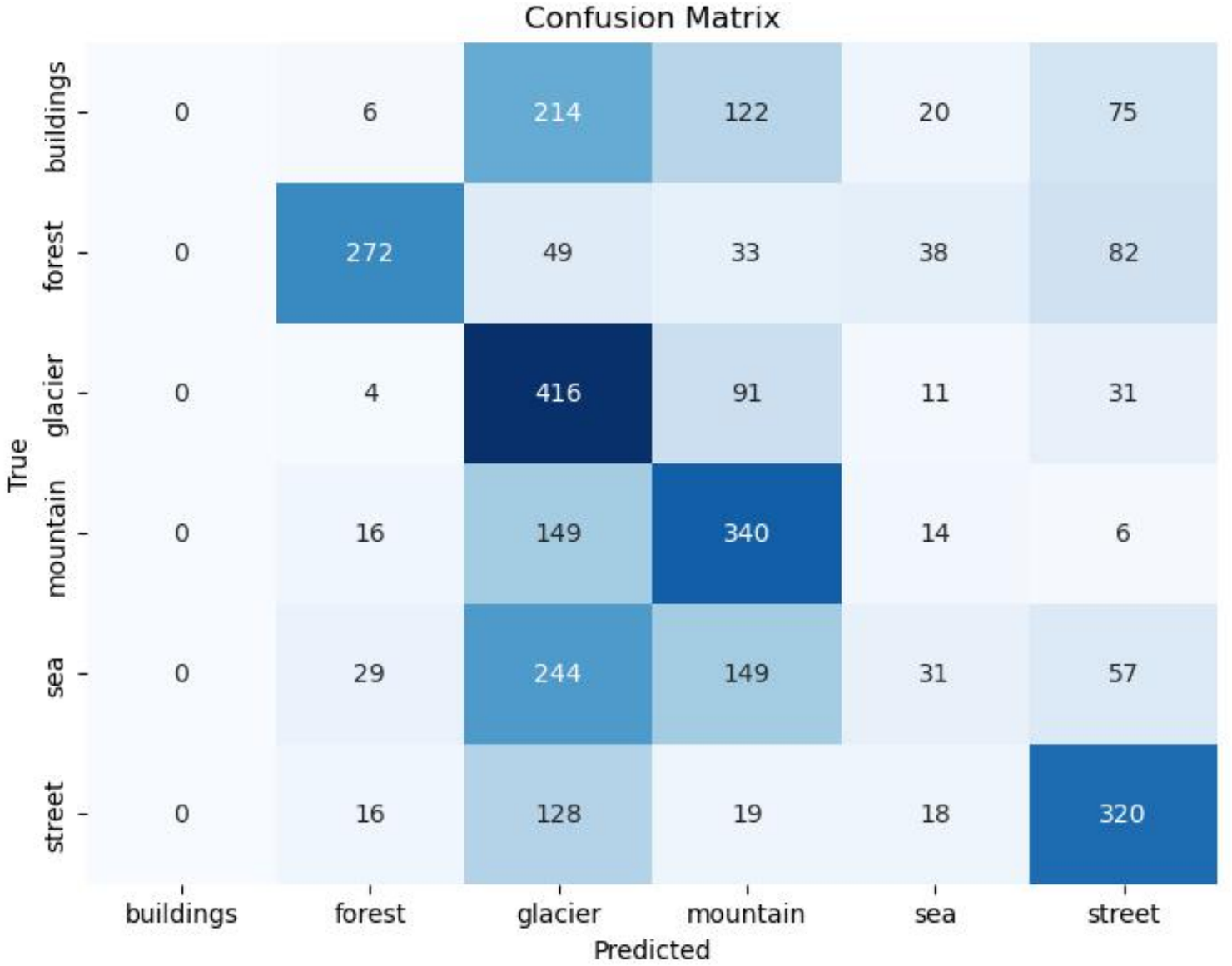


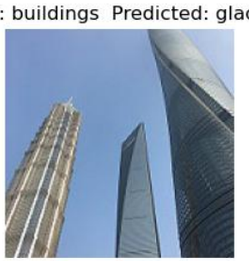


Input Layer (Flatten)
Hidden Layer (Dense with 128 neurons)
Output Layer (Dense with 6 neurons)

Baseline Model (Neural network)

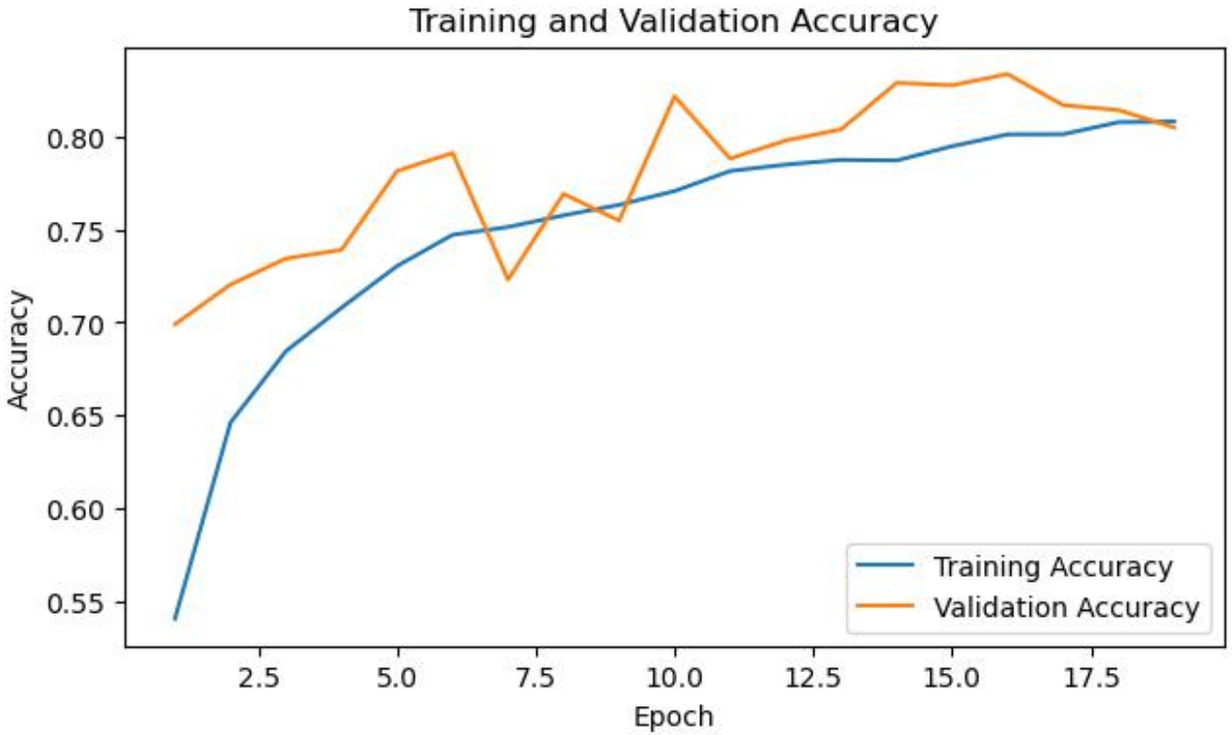
Test: 0.46
Train: 0.43





Model 2 (3 Convolutional Layers)

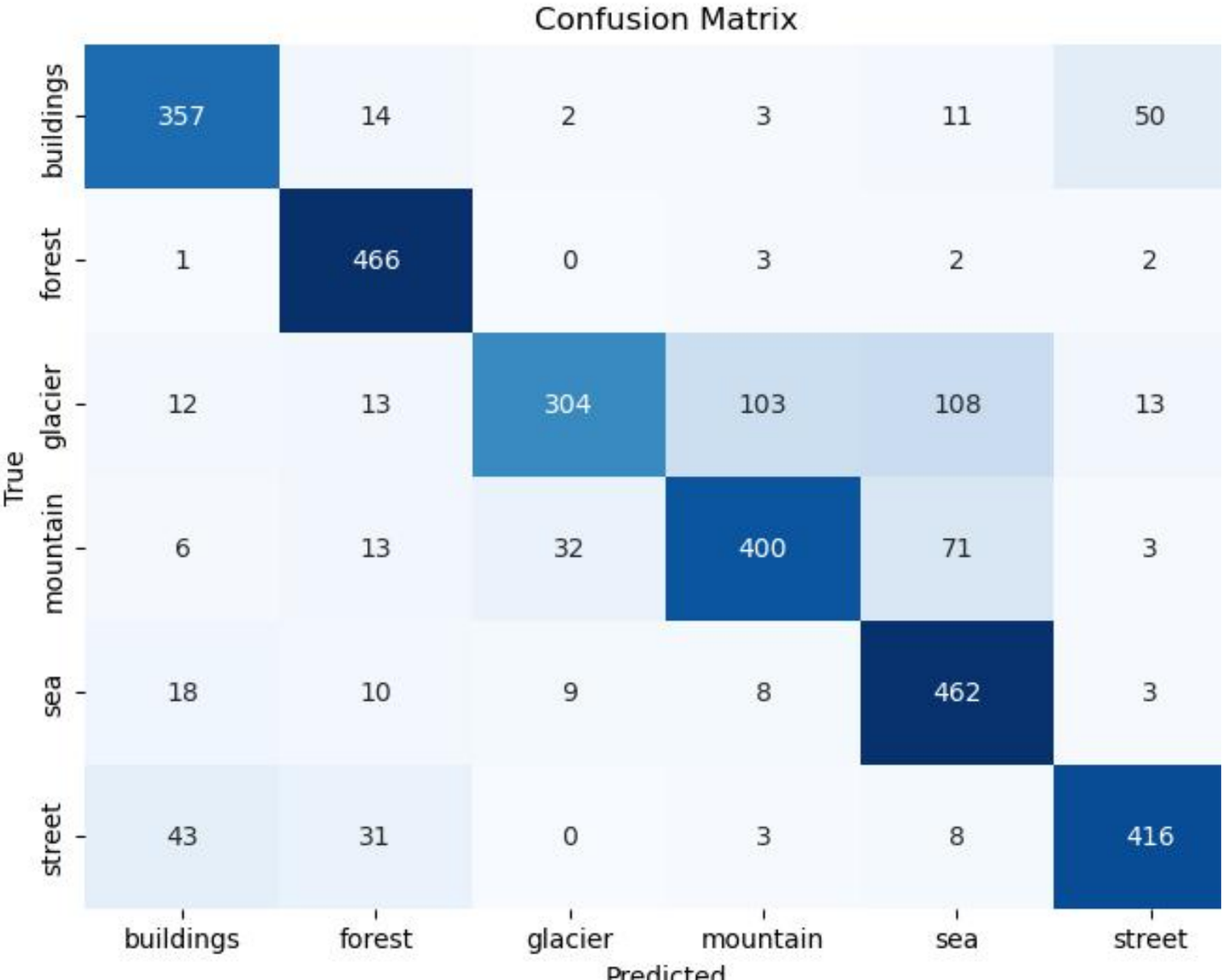
Test: 0.80
Train: 0.81



Conv2D(filters=32, kernel_size=(3, 3))
MaxPooling2D(pool_size=(3, 3))
Conv2D(filters=64, kernel_size=(3, 3))
MaxPooling2D(pool_size=(2, 2))
Conv2D(filters=128, kernel_size=(3, 3))
Flatten()
Dense(filters=128)
Dense(filters=6)

Model 2 (2 Convolutional Layers)

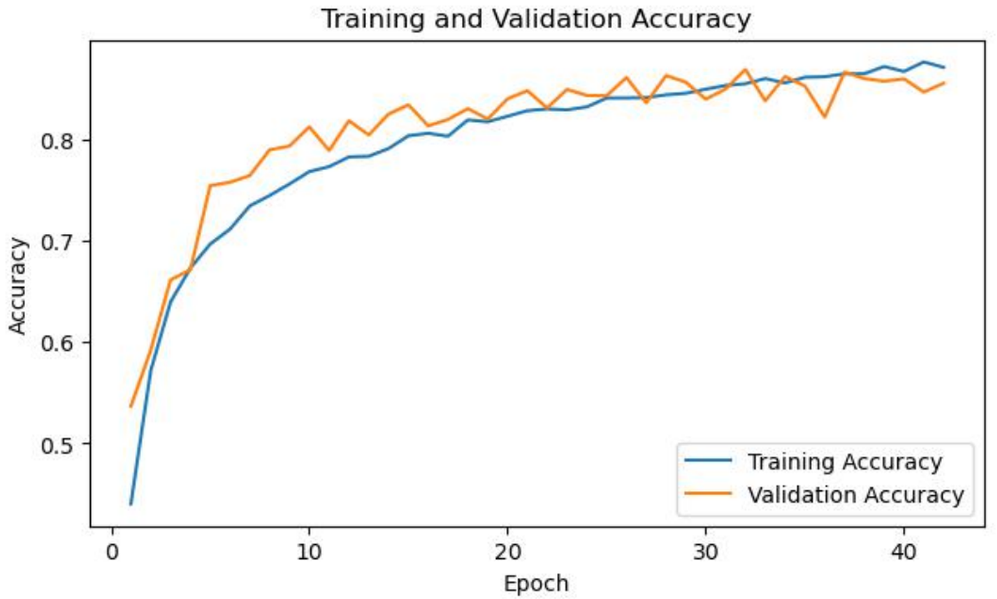
Test: 0.80
Train: 0.81



Model 3 (6 Convolutional Layers)

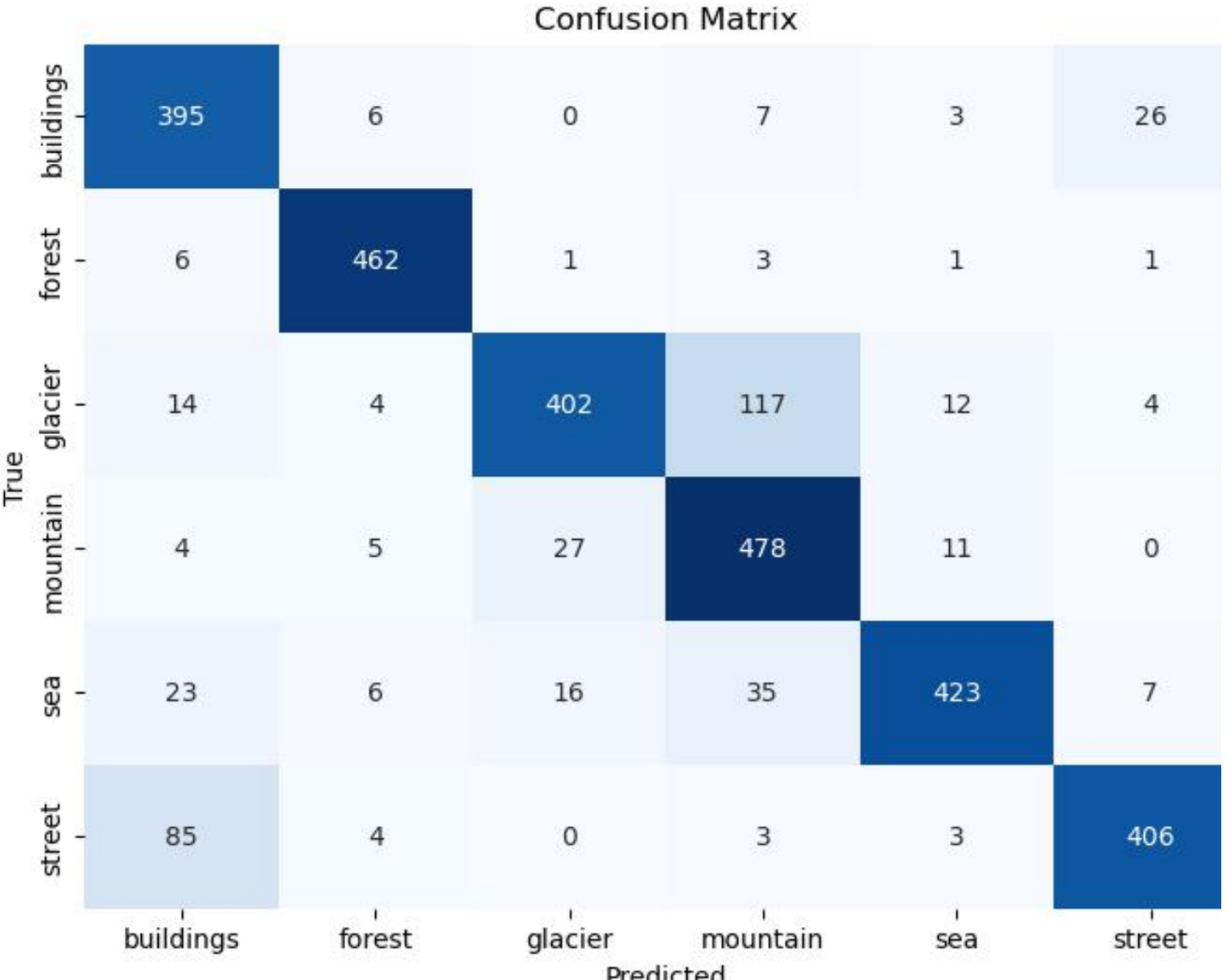
Test: 0.86
Train: 0.87

```
Conv2D(filters=32, kernel_size=(3, 3))
MaxPooling2D(pool_size=(3, 3))
Conv2D(filters=64, kernel_size=(3, 3))
MaxPooling2D(pool_size=(2, 2))
Conv2D(filters=128, kernel_size=(3, 3))
MaxPooling2D(pool_size=(2, 2))
Conv2D(filters=256, kernel_size=(3, 3))
MaxPooling2D(pool_size=(2, 2))
Conv2D(filters=512, kernel_size=(3, 3))
Conv2D(filters=1024, kernel_size=(3, 3))
Flatten()
Dense(filters=1024)
Dense(filters=6)
```



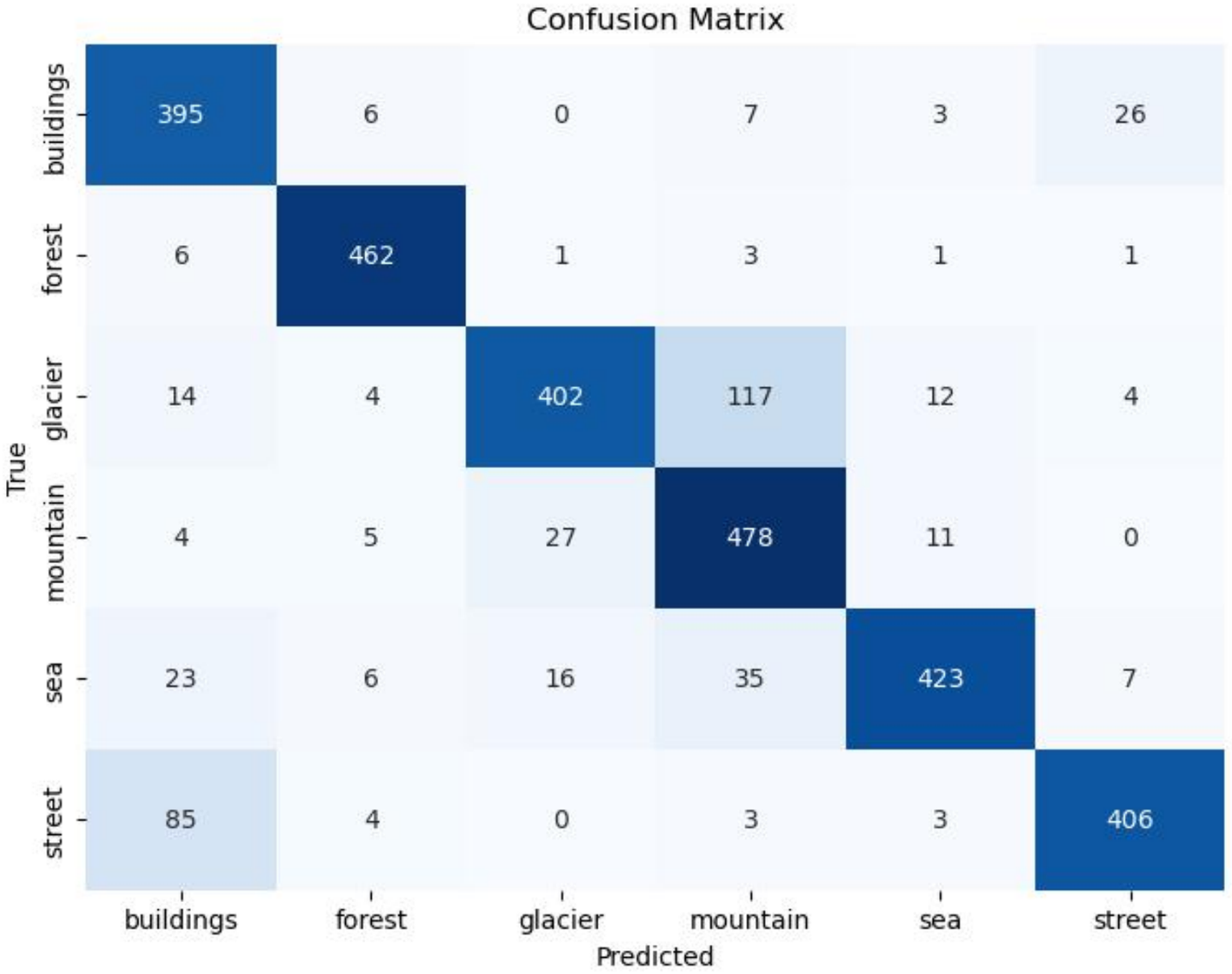
Model 3 (6 Convolutional Layers)

Test: 0.86
Train: 0.87



Model 3 (6 Convolutional Layers)

Test: 0.86
Train: 0.87



Mountain or Galcier?

Street or Building?

Best performing model

Baseline

Fully connected neural network

Test: 0.46
Train: 0.43

Model 2

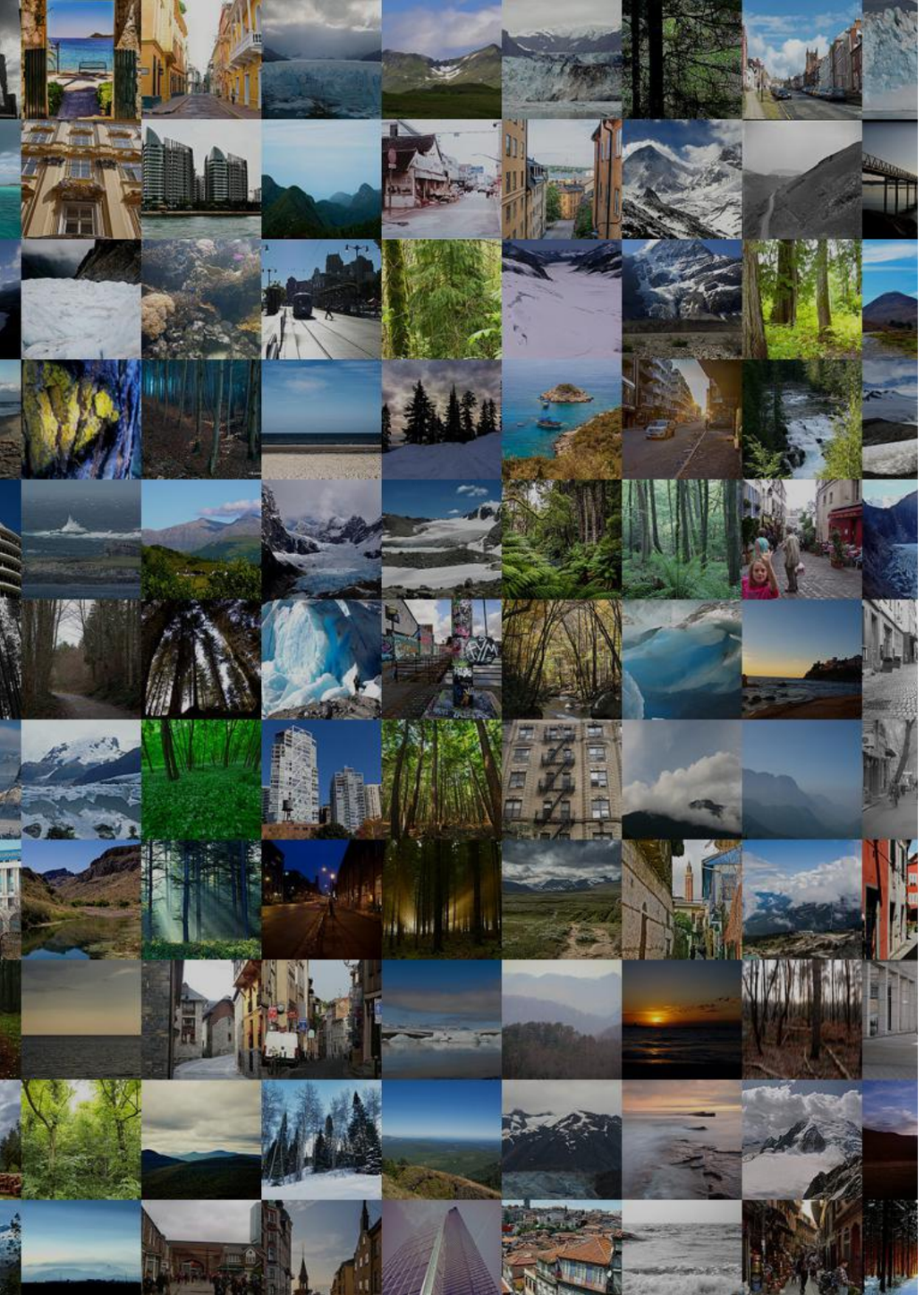
3 Convolutional layers

Test: 0.80
Train: 0.81

Model 3

6 Convolutional layers

Test: 0.86
Train: 0.87



Recommendations

- Begin the implementation of the image classification model for archiving purposes.
- Establish a feedback loop with archivists and users to continually improve the model's performance.
- Explore partnerships and collaboration opportunities with organizations or platforms in related fields, such as content management or digital libraries.

Next steps

create an object detection or scene understanding model that would recognise all the components of the scene even if it included more than one element (Street, building)

Github.com/alihijazy/

Linkedin.com/in/alihijazy/

THANK YOU