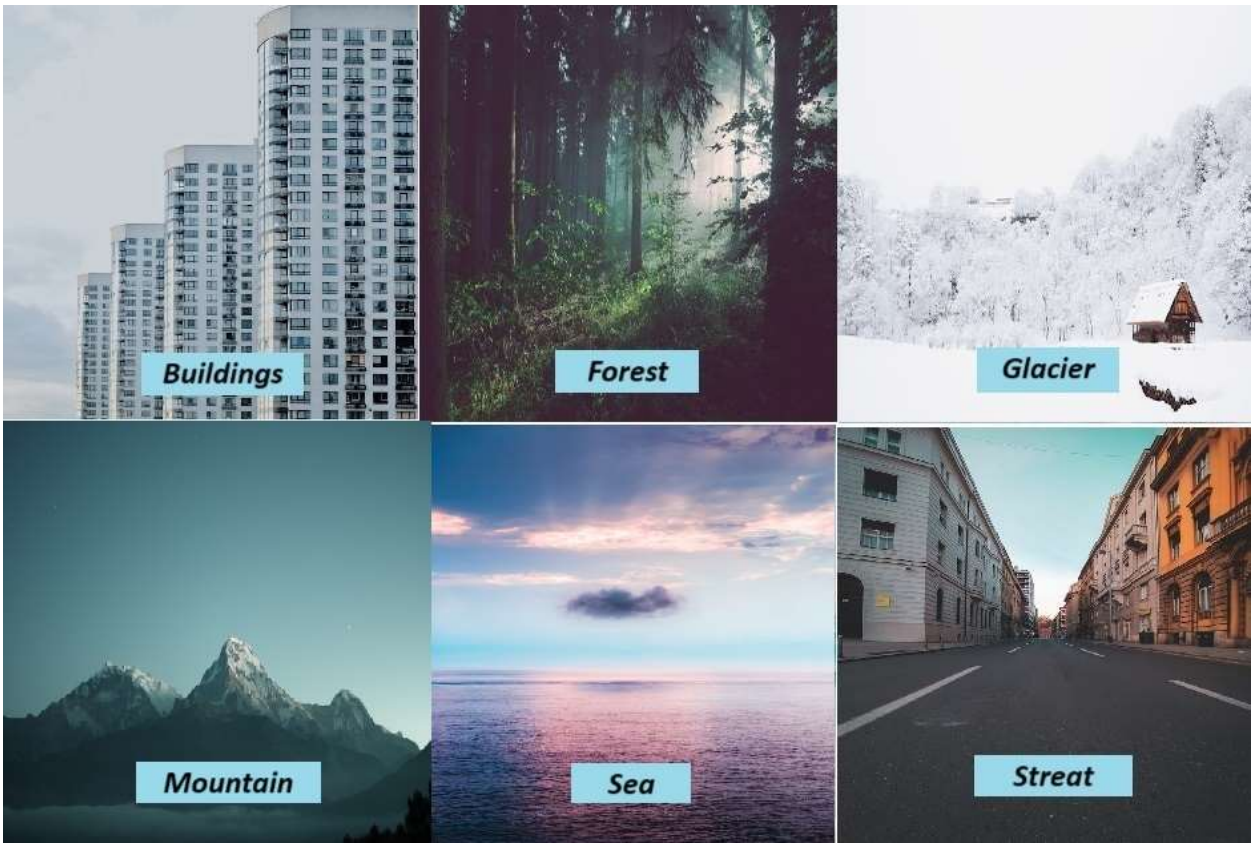
Project Report: Image Classification Using CNN and Transfer Learning



Reported by: Ahmed Abdelnaby Helal

Date: 31-1-2025

Abstract

This project focuses on image classification using Convolutional Neural Networks (CNNs) and transfer learning techniques to predict images across six classes: (Mountain, Buildings, Street, Sea, Glacier, and Forest). A custom CNN model was developed (called 'My-Net'), and transfer learning was applied using pre-trained architectures such as VGG16, VGG19, InceptionV3, ResNet152V, and DenseNet201. Two voting ensembles and two average ensembles were constructed to improve performance. The best-performing model was an average ensemble combining the My-Net, Res-Net, and Dense-Net, achieving 94% accuracy, 94% precision, 94% recall, and 94% F1-score on the test dataset. This report details the methodology, results, and conclusions, including confusion matrices, learning curves, and classification reports for each model.

Table of Contents

List of figures

List of tables

1. Introduction

Image classification is a fundamental challenge in the field of computer vision, with a variety of real-world applications that impact our daily lives. From helping self-driving cars navigate safely to assisting doctors in diagnosing medical conditions through imaging, the importance of accurate image classification cannot be overstated. Over the years, deep learning—particularly through the use of Convolutional Neural Networks (CNNs)—has revolutionized this area, enabling us to achieve impressive accuracy in recognizing and categorizing visual data.

In this project, we set out to classify images into six distinct categories: Mountain, Buildings, Street, Sea, Glacier, and Forest. To do this, we built a custom CNN from the ground up, but we also explored the benefits of transfer learning. This technique allows us to take advantage of pre-trained models that have already learned from extensive datasets like ImageNet, adapting them to our specific task. This approach not only saves time and computational resources but often results in better performance, especially when we have limited data to work with.

We also experimented with ensembling techniques, which involve combining the predictions from multiple models to harness their individual strengths and enhance overall performance. Our results demonstrate the effectiveness of both transfer learning and ensembling, showcasing their ability to deliver highly accurate and reliable predictions for image classification tasks.

2. Methodology

2.1 Dataset

- The dataset consists of images divided into six classes: Mountain, Buildings, Street, Sea, Glacier, and Forest.
- The dataset was split into training, validation, and test sets (e.g., 14,034 training images, 3,000 testing images).
- Data augmentation techniques (e.g., rotation, flipping, zooming) were applied to increase dataset diversity and prevent overfitting.

2.2 Custom CNN Architecture

- The custom CNN model was designed with multiple convolutional layers, max-pooling layers, dropout layers, and fully connected layers.
- Activation functions such as ReLU were used, and the output layer utilized SoftMax for multi-class classification.

2.3 Transfer Learning Models

- Pre-trained models (VGG16, VGG19, Inception, Res-Net, Dense-Net) were fine-tuned for the specific classification task.
- The final fully connected layers of each pre-trained model were replaced with other fully connected layers as well as a new output layer tailored to the six-class problem.
- Models were trained using a combination of frozen and unfrozen layers to balance training time and performance.

2.4 Ensembling Techniques

- **Voting Ensembles**: Two voting ensembles were created:
    1. A hard voting ensemble combining predictions from all models.
    2. A hard voting ensemble combining predictions from three models (custom CNN, Res-Net, Dense-Net).
- **Average Ensembles**: Two average ensembles were created:
    1. An average ensemble combining predictions from all models.
    2. An average ensemble combining predictions from the top three models (custom CNN, Res-Net, Dense-Net).

2.5 Evaluation Metrics

- Models were evaluated using accuracy, precision, recall, and F1-score.
- Confusion matrices were generated to visualize classification performance.
- Training and validation learning curves were plotted to analyze model convergence and overfitting.

3. Results

3.1 Performance of Individual Models

The following table summarizes all the results from all models:

Table 1 Models Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score
My-Net	90%	90%	90%	90%
VGG-16	88%	88%	88%	88%
VGG-19	88%	88%	88%	88%
Inception_V3	91%	91%	91%	91%
ResNet152V2	91%	91%	91%	91%
DenseNet201	92%	92%	92%	92%
Voting Ensemble (All models)	92%	92%	92%	92%
Voting Ensemble (Top 3 models)	93%	93%	93%	93%
Average Ensemble (All models)	93%	93%	93%	93%
Average Ensemble (Top 3 models)	94%	94%	94%	94%

3.2 Performance of Ensembles

- **Voting Ensembles**: Using Voting Ensembles seems to have only a slight increase of performance over the best single model.
- **Average Ensembles**: Using Averaging Ensembles seems to have a good improvement of the performance over the best single model.

3.3 Confusion Matrices

- Confusion matrices for each model and ensemble were generated to visualize true vs. predicted labels.
- The Classes are as follows:

Table 2 Labels Corresponding to Classes names

Buildings	Forest	Glacier	Mountain	Sea	Street
0	1	2	3	4	5

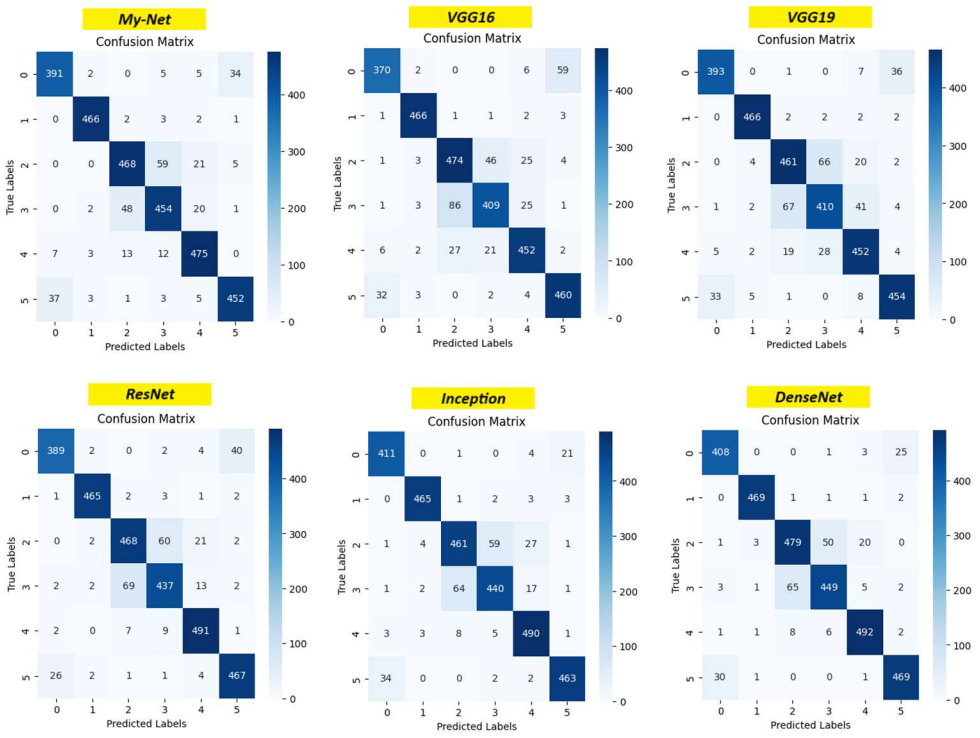


Figure 1 Models Confusion Matrices

The confusion matrices highlight the strengths and weaknesses of each model, providing valuable insights into their classification behavior. While all models perform well on certain classes (e.g., Forest, Buildings and Street), they struggle with others (e.g., Mountain and Glacier). DenseNet emerges as the strongest individual model, but the ensemble approach (as demonstrated in the project) effectively combines the strengths of multiple models to achieve the best overall performance.

1. <u>Glacier (2) and Mountain (3):</u>
    - There is some confusion between **Glacier (2)** and **Mountain (3)**. This could be due to visual similarities between glaciers and mountains (e.g., a lot of mountains have ice on their peaks due to low temperatures at the peak of mountains).
2. <u>Buildings (0) and Street (5):</u>
    - There is some confusion between **Buildings (0)** and **Street (5)**. This could be due to the fact that most streets contain also buildings which can cause some confusion even for a human being classifying images, unless a clear criterion is set for labeling.
3. <u>Best Performing Model:</u>
    - **DenseNet** stands out as the best-performing individual model, with high accuracy across most classes and fewer misclassifications overall.

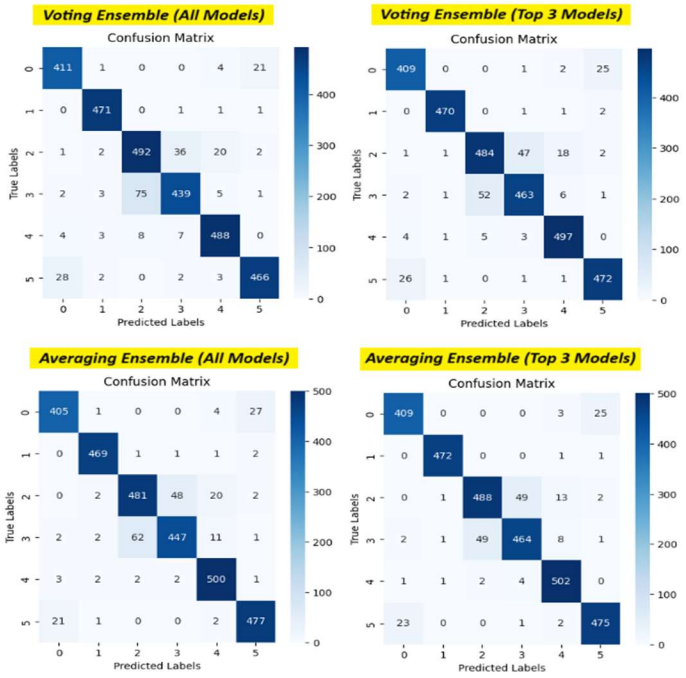


Figure 2 Ensemble Models Confusion Matrices

The **average ensemble** of the top three models (custom CNN, ResNet, and DenseNet) emerged as the best-performing approach. This ensemble outperformed individual models by effectively combining their strengths and mitigating their weaknesses. For instance, while single models like DenseNet performed well, they still struggled with certain classes (e.g., Mountain and Glacier). The ensemble approach reduced misclassifications, particularly in challenging classes, by leveraging the diverse predictions of the constituent models. This highlights the advantage of ensembling in improving robustness and overall classification performance compared to relying on a single model.

3.4 Learning Curves

- Training and validation learning curves were plotted for each model to analyze convergence and overfitting.

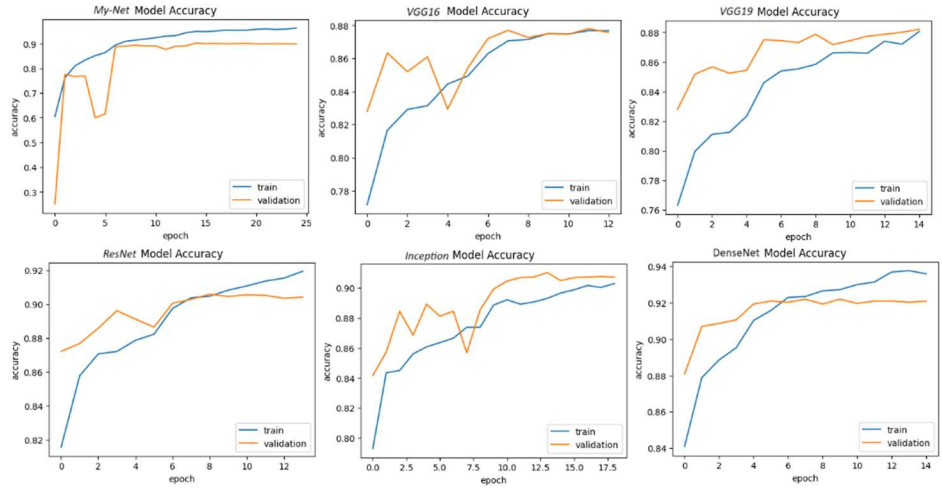


Figure 3 Models Accuracy Learning Curves

- My-Net, VGG16, VGG19: All three models show a trend of increasing accuracy with training epochs. Validation accuracy also increases initially but then stabilize for all models.
- ResNet, Inception, DenseNet: All three models show a clear trend of increasing accuracy with training epochs, and validation accuracy remains stable or increases slightly.

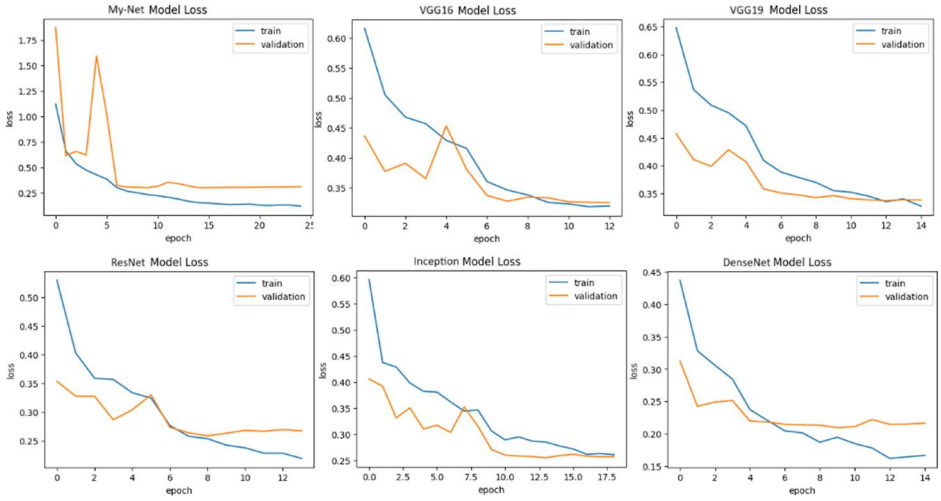


Figure 4 Models Loss Learning Curves

- My-Net: Shows a trend of decreasing loss with training epochs. Validation loss also decreases but fluctuates in the start then stabilizes.
- VGG16, VGG19, ResNet, Inception, DenseNet: All five models show a clear trend of decreasing loss with training epochs, and validation loss also decreases steadily while having small fluctuations.
- To prevent overfitting, we used Early callback, Reduce learning rate on plateau, Data augmentation.

3.5 Classification Reports

- Detailed classification reports (precision, recall, F1-score, support) were generated for each model and ensemble, which can be seen in the attached appendix (A).

3.6 Prototype Application

- A prototype application was developed using **QT5** to demonstrate the practical implementation of the image classification system. The application provides a user-friendly interface for classifying images into six categories: **Mountain**, **Buildings**, **Street**, **Sea**, **Glacier**, and **Forest**. Users can upload images, select from a variety of models (including **My-Net**, **VGG16**, **VGG19**, **ResNet**, **Inception**, **DenseNet**, and an ensemble of models), and receive real-time predictions. Key features of the application include:
- **Image Upload and Info**: Users can upload images in common formats (e.g., JPEG, PNG), and the application displays image details such as dimensions (e.g., 800 x 640).
- **Model Selection**: Users can choose from individual models or the ensemble, with a note that the ensemble prediction may take longer due to its computational complexity.
- **Edge Detection**: The application includes an edge detection feature (e.g., Canny edges) to provide additional visual insights into the uploaded image.
- **Real-Time Prediction**: After processing, the application displays the predicted class (e.g., "Buildings") along with the confidence score (e.g., 99.92%).
- **Clear and Intuitive Interface**: The results are presented in a clean and visually appealing manner, with options to clear the results and start a new prediction.
- This prototype highlights the versatility and practicality of the image classification system, showcasing its potential for real-world applications such as environmental monitoring, urban planning, and tourism. Future enhancements could include support for additional classes, integration of more advanced image processing features, and optimization for mobile platforms to increase accessibility.

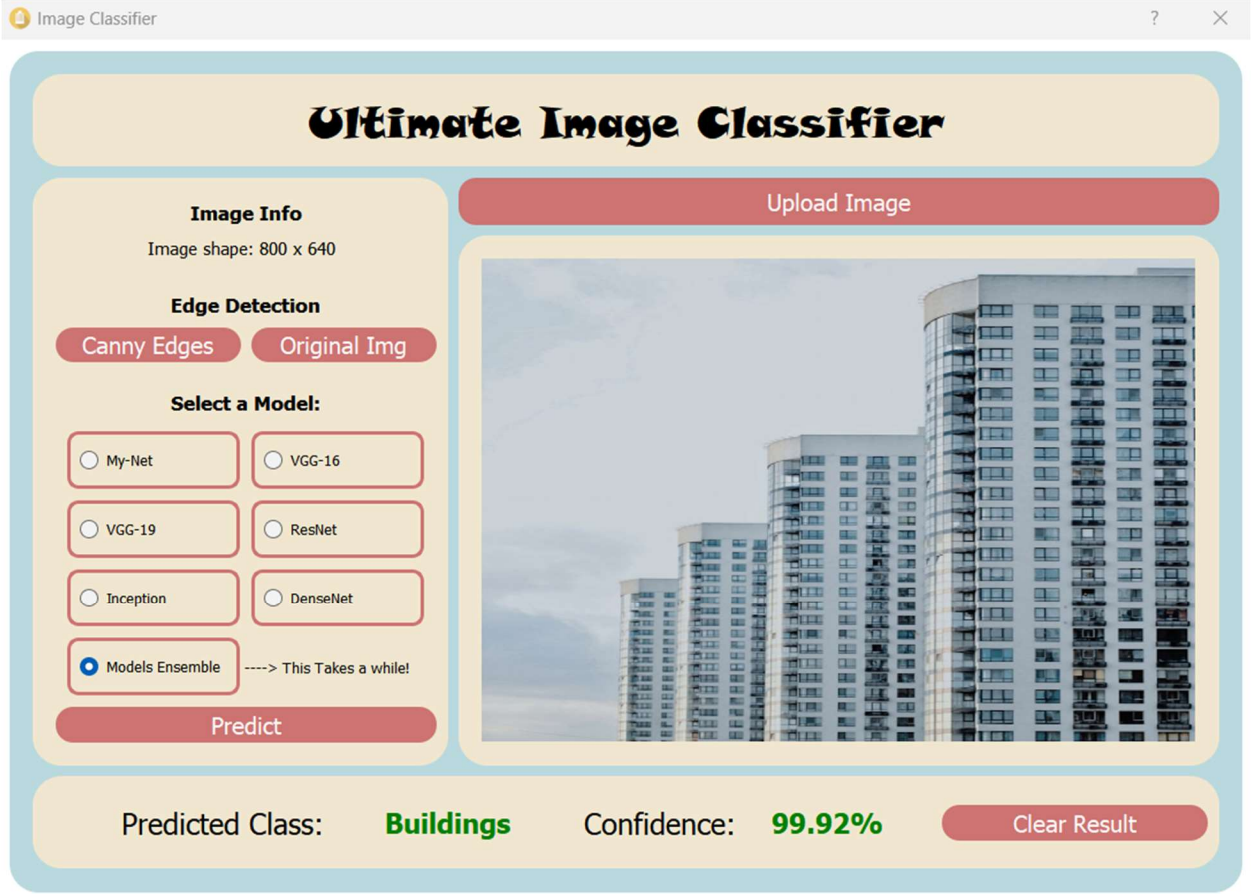


Figure 5 Application Prototype

4. Discussion

- The custom CNN performed well but was outperformed by transfer learning models, highlighting the benefits of leveraging pre-trained architectures.
- ResNet and DenseNet achieved the highest individual performance, likely due to their deeper architectures and skip connections.
- The average ensemble of the top three models (custom CNN, ResNet, DenseNet) achieved the best performance (94% accuracy), demonstrating the effectiveness of combining diverse models.
- Voting ensembles performed slightly worse than average ensembles, possibly due to the diversity in model predictions.
- Confusion matrices revealed specific classes (e.g., Glacier vs. Sea) where models struggled, indicating potential areas for improvement.
- Learning curves showed that most models converged well, with minimal overfitting due to data augmentation and dropout layers.

5. Conclusion

This project successfully applied CNNs and transfer learning to classify images into six categories. The best-performing model was an average ensemble combining the custom CNN, ResNet, and DenseNet, achieving 94% accuracy, precision, recall, and F1-score. Transfer learning models consistently outperformed the custom CNN, demonstrating the value of leveraging pre-trained architectures. Ensembling further improved performance by combining the strengths of individual models. Future work could explore additional data augmentation techniques, hyperparameter tuning, and testing other pre-trained models like EfficientNet or MobileNet.

APPENDIX (A)

This Appendix includes all the classification reports for all models trained in this project.

Classification Report:

	precision	recall	f1-score	support
buildings	0.90	0.89	0.90	437
forest	0.98	0.98	0.98	474
glacier	0.88	0.85	0.86	553
mountain	0.85	0.86	0.86	525
sea	0.90	0.93	0.92	510
street	0.92	0.90	0.91	501
accuracy			0.90	3000
macro avg	0.90	0.90	0.90	3000
weighted avg	0.90	0.90	0.90	3000

Precision: 0.9020
Recall: 0.9020
F1-score: 0.9019

Figure 6 My-Net Classification Report

Classification Report:

	precision	recall	f1-score	support
buildings	0.90	0.85	0.87	437
forest	0.97	0.98	0.98	474
glacier	0.81	0.86	0.83	553
mountain	0.85	0.78	0.81	525
sea	0.88	0.89	0.88	510
street	0.87	0.92	0.89	501
accuracy			0.88	3000
macro avg	0.88	0.88	0.88	3000
weighted avg	0.88	0.88	0.88	3000

Precision: 0.8776
Recall: 0.8770
F1-score: 0.8766

Figure 7 VGG16 Classification Report

```
Classification Report:
              precision    recall  f1-score   support

   buildings       0.91      0.90      0.90       437
      forest       0.97      0.98      0.98       474
     glacier       0.84      0.83      0.84       553
    mountain       0.81      0.78      0.80       525
         sea       0.85      0.89      0.87       510
      street       0.90      0.91      0.91       501

    accuracy                           0.88      3000
   macro avg       0.88      0.88      0.88      3000
weighted avg       0.88      0.88      0.88      3000

Precision: 0.8783
Recall: 0.8787
F1-score: 0.8784
```

Figure 8 VGG19 Classification Report

```
Classification Report:
              precision    recall  f1-score   support

   buildings       0.93      0.89      0.91       437
      forest       0.98      0.98      0.98       474
     glacier       0.86      0.85      0.85       553
    mountain       0.85      0.83      0.84       525
         sea       0.92      0.96      0.94       510
      street       0.91      0.93      0.92       501

    accuracy                           0.91      3000
   macro avg       0.91      0.91      0.91      3000
weighted avg       0.91      0.91      0.91      3000

Precision: 0.9054
Recall: 0.9057
F1-score: 0.9053
```

Figure 9 ResNet Classification Report

Classification Report:

	precision	recall	f1-score	support
buildings	0.91	0.94	0.93	437
forest	0.98	0.98	0.98	474
glacier	0.86	0.83	0.85	553
mountain	0.87	0.84	0.85	525
sea	0.90	0.96	0.93	510
street	0.94	0.92	0.93	501
accuracy			0.91	3000
macro avg	0.91	0.91	0.91	3000
weighted avg	0.91	0.91	0.91	3000

Precision: 0.9097
Recall: 0.9100
F1-score: 0.9095

Figure 10 Inception Classification Report

Classification Report:

	precision	recall	f1-score	support
buildings	0.92	0.93	0.93	437
forest	0.99	0.99	0.99	474
glacier	0.87	0.87	0.87	553
mountain	0.89	0.86	0.87	525
sea	0.94	0.96	0.95	510
street	0.94	0.94	0.94	501
accuracy			0.92	3000
macro avg	0.92	0.92	0.92	3000
weighted avg	0.92	0.92	0.92	3000

Precision: 0.9217
Recall: 0.9220
F1-score: 0.9218

Figure 11 DenseNet Classification Report

Classification Report:

	precision	recall	f1-score	support
buildings	0.92	0.94	0.93	437
forest	0.98	0.99	0.99	474
glacier	0.86	0.89	0.87	553
mountain	0.91	0.84	0.87	525
sea	0.94	0.96	0.95	510
street	0.95	0.93	0.94	501
accuracy			0.92	3000
macro avg	0.92	0.92	0.92	3000
weighted avg	0.92	0.92	0.92	3000

Precision: 0.9225
Recall: 0.9223
F1-score: 0.9221

Figure 12 Voting Ensemble (All Models) Classification Report

Classification Report:

	precision	recall	f1-score	support
buildings	0.93	0.94	0.93	437
forest	0.99	0.99	0.99	474
glacier	0.89	0.88	0.88	553
mountain	0.90	0.88	0.89	525
sea	0.95	0.97	0.96	510
street	0.94	0.94	0.94	501
accuracy			0.93	3000
macro avg	0.93	0.93	0.93	3000
weighted avg	0.93	0.93	0.93	3000

Precision: 0.9313
Recall: 0.9317
F1-score: 0.9314

Figure 13 Voting Ensemble (Top 3 Models) Classification Report

Classification Report:

	precision	recall	f1-score	support
buildings	0.94	0.93	0.93	437
forest	0.98	0.99	0.99	474
glacier	0.88	0.87	0.88	553
mountain	0.90	0.85	0.87	525
sea	0.93	0.98	0.95	510
street	0.94	0.95	0.94	501
accuracy			0.93	3000
macro avg	0.93	0.93	0.93	3000
weighted avg	0.93	0.93	0.93	3000

Precision: 0.9259
Recall: 0.9263
F1-score: 0.9259

Figure 14 Averaging Ensemble (All Models) Classification Report

Classification Report:

	precision	recall	f1-score	support
buildings	0.94	0.94	0.94	437
forest	0.99	1.00	0.99	474
glacier	0.91	0.88	0.89	553
mountain	0.90	0.88	0.89	525
sea	0.95	0.98	0.97	510
street	0.94	0.95	0.95	501
accuracy			0.94	3000
macro avg	0.94	0.94	0.94	3000
weighted avg	0.94	0.94	0.94	3000

Precision: 0.9363
Recall: 0.9367
F1-score: 0.9364

Figure 15 Averaging Ensemble (Top 3 models) Classification Report