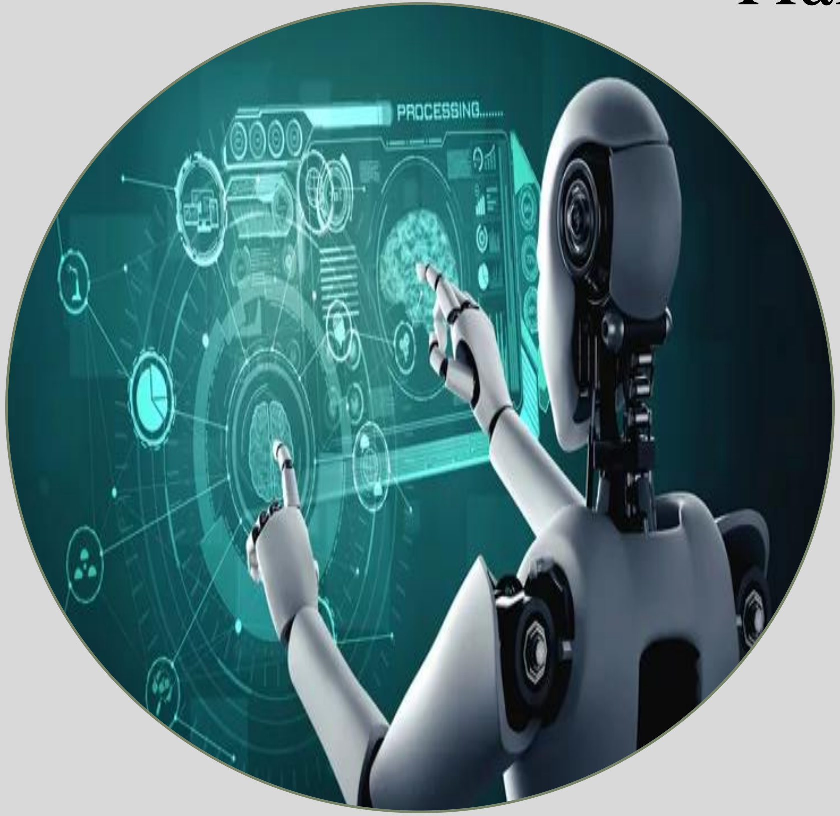
"Transfer Learning in Image Classification"

Harnessing Pre-trained Models for Advanced Image Recognition



Presented by: Group 29

Team Members:
Prudhvi Vardhan - 22071880
Medi Praharsh Vijay – 22098361
A. Dhanalakshmi Shilpa - 22024537
Pavan Kumar Kurva – 22065713
Harishchandra chunchu-22065578
Abhishek Aeera-22097107
Tati Machindra - 22076527

Presentation Date: 18 March

Google Colab link:
https://colab.research.google.com/drive/1or_iiRKke1SBUwxUokgN96OZVXXMnlrT#scrollTo=81445993

Overview of the Project

- In our project, we've undertaken a multi-class image classification task starting with the construction of a base CNN model tailored to our data.

- This bespoke model served as a benchmark, showcasing the potential of CNNs in distinguishing complex patterns in image data. However, the model's performance, as revealed by the training curves and confusion matrix, indicated a propensity for overfitting beyond a certain number of epochs.

- This challenge often arises with custom models, especially when the available training data is limited or lacks the diversity needed to generalize well to unseen examples.

- To address this, we employed transfer learning with the VGG16 architecture, a model renowned for its performance on the ImageNet dataset. Initially, you froze the pre-trained model's layers, allowing your network to leverage the rich feature detectors learned from a vast array of images.

- By keeping the weights static in the initial training phase, our model could maintain the integrity of these features, which are often universally applicable across various image recognition tasks. As the training progressed, we selectively unfroze layers and fine-tuned the model to align its understanding more closely with our specific dataset.

- This approach, coupled with fine-grained hyper parameter tuning such as adaptive learning rates and early stopping, yielded a significant boost in accuracy and generalization, as evidenced by the improved loss metrics and a more precise confusion matrix. The strategic unfreezing of layers allowed the VGG16 model to refine its pre-learned representations to better fit the nuances of our data, thereby enhancing the model's predictive prowess

Introduction to the Data set

• **Description:**

- The Natural Scenes dataset comprises over 14,000 images, meticulously categorized into various natural and urban scenes.
- Each image typically has a resolution of 150x150 pixels, ensuring compatibility with machine learning models while minimizing computational resource requirements.
- The dataset includes categories such as buildings, forests, glaciers, mountains, seas, and streets. These categories are essential for tasks involving the classification and differentiation of different types of natural and man-made environments.

• **Dataset Composition:**

- **Images**: Over 14,000 images, representing diverse natural and urban scenes.
- **Resolution**: Typically 150x150 pixels, facilitating model training without excessive computational overhead.

• **Categories:**

- The dataset encompasses various categories such as buildings, forests, glaciers, mountains, seas, and streets.
- These categories are crucial for tasks requiring the identification and classification of different types of environments.

• **Dataset Segmentation:**

- **Training Set:** Contains a substantial portion of the dataset, utilized for training machine learning models.
- **Test Set:** Comprises a smaller subset used for evaluating model performance and generalization capabilities.
- **Prediction Set:** Possibly designated for real-world testing or competitions, where model predictions are assessed under diverse scenarios.

• **Significance:**

- The dataset's segmentation into training, test, and prediction sets facilitates systematic model development, evaluation, and real-world deployment.
- Its diverse composition and segmentation strategy make it a valuable resource for training, testing, and benchmarking image classification models across various applications and domains.

Selection of the pre-trained model and its original purpose

Introduction:

- In this report, we will discuss the selection of the VGG16 pre-trained model for image classification tasks, providing an overview of its original purpose and comparing it with other popular architectures like ResNet and Inception. VGG16, developed by the Visual Geometry Group at the University of Oxford, has garnered widespread recognition for its simplicity and effectiveness in feature extraction from images.

Original Purpose of VGG16:

- The VGG16 model was initially developed for image classification tasks, with its primary purpose being participation in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Trained on the ImageNet dataset, which contains over a million labeled images across 1,000 categories, VGG16 demonstrated outstanding performance in categorizing diverse visual content.

Factors for Selection of Pre-trained Model: VGG16

1. **Simplicity and Effectiveness:** VGG16's simple and uniform architecture makes it easy to understand and implement, making it a popular choice for image classification tasks.
2. **Proven Performance:** Despite its simplicity, VGG16 has shown remarkable performance in various image classification benchmarks, including ImageNet, demonstrating its effectiveness in extracting features from images.
3. **Compatibility with Dataset:** Given the dataset's composition of various natural and urban scenes, VGG16's ability to learn hierarchical features from images aligns well with the classification task at hand.
4. **Resource Requirements**:
   VGG16 model requires fewer computational resources (CPU, GPU, memory) compared to other training models from scratch, making it more feasible for resource-constrained environments like Resnet, Inception etc..
   From Scratch: Typically demands more computational resources, especially for training deep neural networks from scratch, which may necessitate high-performance hardware.

Justification: The selection of the VGG16 pre-trained model for image classification tasks offers a balance between simplicity, effectiveness, and performance. While architectures like ResNet and Inception offer advancements in handling deeper networks and multi-scale features, it also requires huge computational resources. VGG16's proven track record and straightforward design make it a suitable choice for the provided dataset. Leveraging the pre-trained weights of VGG16 enables efficient model development and enhances the performance of image classification systems, contributing to advancements in computer vision research and applications

Overview of transfer learning and its importance

Description:

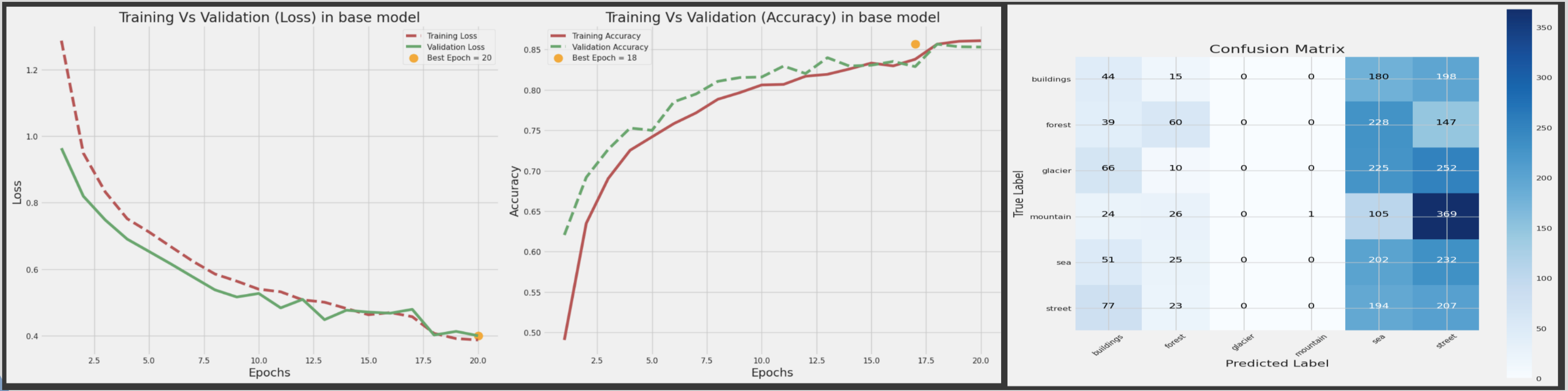
Transfer learning is a machine learning technique where knowledge gained from solving one problem is applied to a different but related problem. In the context of image classification, transfer learning involves leveraging a pre-trained model's learned features and weights and applying them to a new classification task.

1. **Reuse of Pre-trained Models:**
   - Pre-trained models, such as VGG16, ResNet, or Inception, are trained on large-scale image datasets like ImageNet. These models learn generic features and patterns from vast amounts of data during training.
   - Instead of training a model from scratch, transfer learning allows us to use the knowledge encoded in these pre-trained models as a starting point for new tasks.
   - By reusing pre-trained models, we can benefit from the learned representations of images, which capture general features like edges, textures, and object shapes.
2. **Expedited Model Training:**
   - Transfer learning expedites model training by leveraging pre-trained models' learned representations. Instead of starting with random weights, the model starts with weights already fine-tuned on a large dataset.
   - This significantly reduces the computational resources and time required for training, making transfer learning particularly advantageous, especially when working with limited data.
   - Transfer learning allows us to achieve good performance even with smaller datasets, as the model can adapt its learned features to the new task more efficiently.
3. **Improved Performance with Limited Data:**
   - One of the significant advantages of transfer learning is its ability to improve performance, especially in scenarios where the target dataset is small.
   - By initializing the model with pre-trained weights, the model can capture generalizable features from the source task and fine-tune them to better suit the nuances of the target task.
   - This adaptation process enables the model to achieve higher accuracy and better generalization on the target task, even with limited labeled data.
4. **Significance in Image Classification:**
   - In image classification, where labeled data can be scarce and expensive to obtain, transfer learning plays a crucial role in overcoming data limitations.
   - It allows researchers and practitioners to build robust and accurate image classification models by leveraging the knowledge encoded in pre-trained models and adapting it to specific classification tasks.

Basic CNN Model & Results

1.Training from Scratch:

- **Model Architecture:** Utilized a simple CNN architecture consisting of convolutional and max-pooling layers followed by dense layers.
- **Training Metrics:** After **20** epochs, the training accuracy was approximately **86.09%**, while the validation accuracy was **85.31%**.
- **Testing Performance:** The model's performance on the test set was poor, achieving a test accuracy of only **17.13%**. This indicates significant overfitting during training.



Loss and Accuracy Curves: The training loss decreases sharply and then continues to decline at a slower rate, indicating good learning. However, the validation loss slightly increases after **epoch 12,** hinting at overfitting.

For accuracy: the validation accuracy surpasses training accuracy early and then both converge, which is a good sign, though a hint of overfitting is also observed as training continues past the best epoch.

Best Epoch: The best epoch based on validation loss is marked at **epoch 20** for loss and **epoch 18** for accuracy. This indicates a slight divergence in the epoch that minimizes loss versus maximizes accuracy.

Confusion Matrix: The base model seems to struggle with certain classes, especially distinguishing between buildings and streets, with a significant number of misclassifications.

VGG16 Model Results

Transfer Learning with VGG16:

- **Model Architecture:** Used the pre-trained VGG16 model as the base architecture and added custom dense layers for classification.
- **Training Metrics:** Initially After **25 epochs**, the training accuracy reached **86.23%**, while the validation accuracy was **85.77%**.
- **Testing Performance:** The model performed significantly better on the test set compared to training from scratch, achieving a test accuracy of **92%** after fine tunning.

Training vs. Validation Loss:

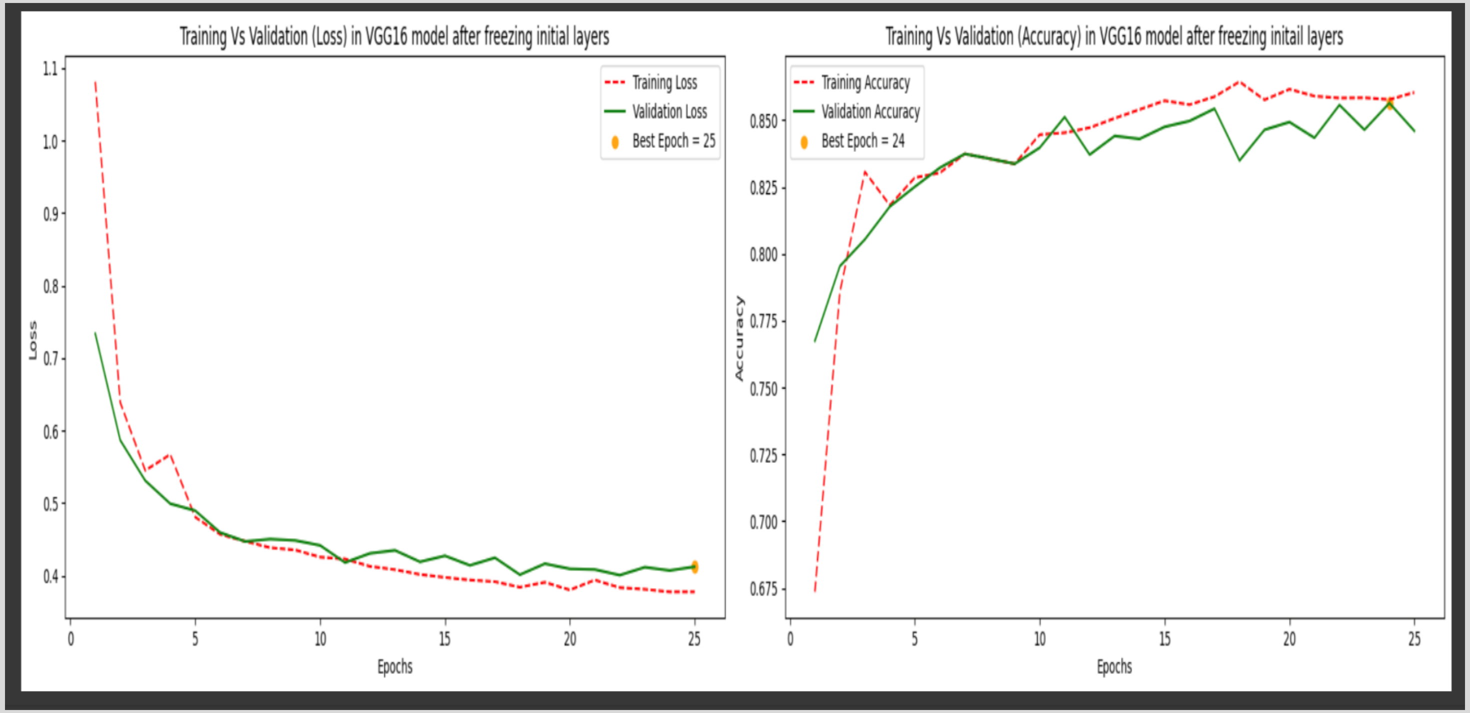
- The loss graphs show that the training loss starts high but decreases rapidly, which is a good sign of the model learning from the data.
- The validation loss initially follows the training loss closely and then starts to plateau, indicating that the model is starting to generalize well without overfitting significantly.
- The best validation loss recorded is approximately **0.3776**.

Training vs. Validation Accuracy:

- The accuracy curve for the training set begins at a lower value but quickly climbs, indicating that the model is effectively capturing the patterns in the data.
- The validation accuracy follows a similar upward trend, with a slight divergence from the training accuracy, which is quite common in training dynamics. The best validation accuracy observed is around **85.77%**.

Best Epochs:

- The best epoch, in terms of validation loss, is at **epoch 25**. This is the point where the model achieves the balance between learning from the training data and generalizing to the validation data without overfitting.
- The best epoch for validation accuracy is slightly earlier at **epoch 24**, suggesting that beyond this point, the model may be improving in terms of loss minimization



<u>Freezing All Layers</u>

- Initially, all layers of the VGG16 pre-trained model were frozen to preserve the learned representations from the ImageNet dataset.

- The model's architecture and parameters remained intact, and only the densely connected layers were trained to adapt to the new dataset. Despite limited trainable parameters **(531,462)**, the model achieved notable performance

Fine-tuning Process

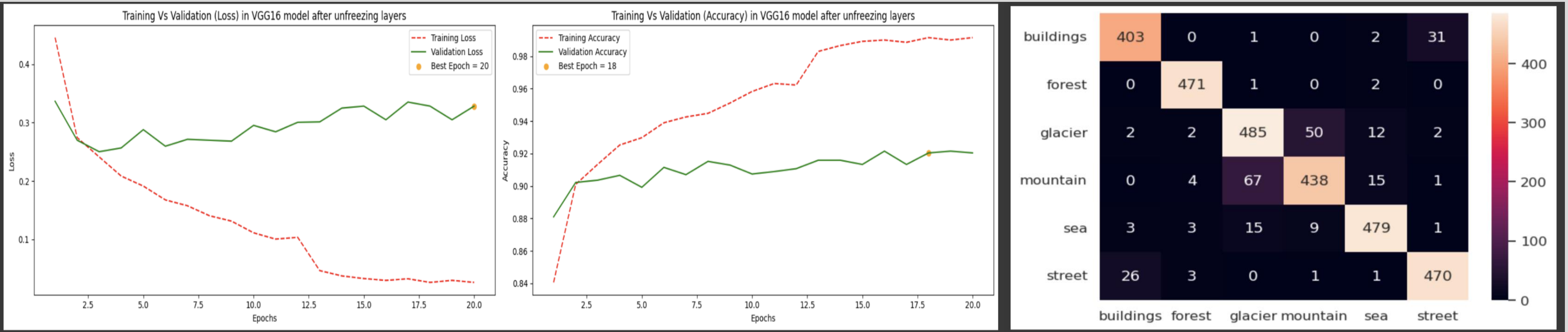
After Unfreezing Layers:

In the final step, we selectively fine-tuned the VGG16 model by freezing 1st 11 initial layers and unfreezing deeper layers for training. This strategy allows the model to leverage both generic and dataset-specific features, potentially leading to enhanced performance. The results were promising:

•**Training & Validation Accuracy:** The training accuracy continued to improve, reaching **99.15%** after **20 epochs**. The validation accuracy also showed improvement, reaching **92.04%.**

Conclusion:

Comparing the performance of different fine-tuning strategies, we observed that selective fine-tuning of the VGG16 model yielded the highest accuracy on the validation set. By unfreezing deeper layers, the model could adapt its representations more effectively to the new dataset, resulting in improved classification performance. These findings highlight the importance of fine-tuning in transfer learning and underscore the effectiveness of leveraging pre-trained models for various tasks.



Loss and Accuracy Curves: After unfreezing the layers, the loss curves are smoother than the base model, and the validation loss(**0.3283**) closely follows the training loss(**0.0263**) without obvious signs of overfitting. Accuracy also improves, and the curves are closer together, showing that the model generalizes better.

Best Epoch: The best epoch based on validation loss is 20, and for accuracy, it is 18. This shows that the model performance continues to improve slightly as training progresses.

Confusion Matrix: Post fine-tuning, the confusion matrix shows much better classification results with higher correct predictions across all classes.

Comparison of results: Transfer learning vs. from Basic model

- In our project, we've developed a base CNN model and compared its performance against a VGG16 model under two conditions: with layers frozen and then with layers unfrozen for fine-tuning.

- The base model's training process reveals a steady decrease in training loss and an increase in training accuracy over epochs.

- However, it shows signs of overfitting, as evidenced by the validation loss beginning to increase after the **12th epoch**, peaking at a best epoch of 20 for loss and **18** for accuracy. The base model's validation accuracy indicates a ceiling effect where it plateaus, suggesting limited learning from the data.

- The confusion matrix for the base model highlights confusion between certain classes, such as buildings and streets, which indicates room for improvement in feature distinction.

- In contrast, the VGG16 model after initial layer freezing and subsequent unfreezing showcases more promising results. With initial freezing, of 1^{st} 11 layers the model achieves a validation loss of approximately **0.3776** and a validation accuracy of about **85.77%** by the **20th epoch**.

- This suggests a well-fitted model with good generalization capabilities, as reflected by the close tracking of validation loss to training loss and less fluctuation in validation accuracy.

- After unfreezing the layers and fine-tuning, the VGG16 model further refines its performance, reaching up to **92%** in validation accuracy.

- These values signify an exceptional leap in learning, with the model effectively tuning its complex, pre-learned features to the nuances of the new dataset, and achieving a high degree of precision in classification across the various categories

Limitations and potential areas of improvement & Ethical Implications

Domain Mismatch: Understanding how crucial it is to match source and target domains in order to guarantee that knowledge transferred is applicable. This emphasizes the necessity of fine-tuning domain-specific data and performing compatibility checks.

Task Specificity: Recognizing that pre-trained models might not be knowledgeable about particular tasks, requiring the creation of domain-specific models or additional fine-tuning to capture particular task nuances.

Limited adaptability refers to the difficulty of adapting to target tasks or datasets in the best possible way, the importance of precisely tuning parameters, and the investigation of adaptive fine-tuning techniques or ensemble methods.

Data Bias and Fairness: To guarantee equitable model predictions across demographic groups, it is necessary to address the biases that are inherited from the source data and to employ mitigating techniques like bias correction and fairness-aware training.

Computational Resources: Fine-tuning large-scale pre-trained models like VGG16 or ResNet can be computationally intensive, requiring substantial GPU resources and time. This poses a challenge, especially for researchers or organizations with limited computational capabilities. Developing lightweight architectures or exploring transfer learning techniques optimized for resource-constrained environments can alleviate this limitation.

Ethical Considerations: Stressing the moral issues surrounding data privacy, ownership, and consent; promoting openness in disclosure; adhering to moral standards; and securing appropriate consent before using data.

Interpretability and Explainability: Recognizing that it can be difficult to understand intricate model architectures, this section offers strategies to improve interpretability and reliability, such as attention mechanisms and model distillation.

Overall, the results highlight the multifaceted nature of transfer learning's constraints and the significance of resolving these issues by combining technological advancements, moral considerations, and transparency initiatives.

Conclusion

- In conclusion, our project deftly illustrates the transformative power of transfer learning in image classification.
- Starting with the base CNN model, we navigated initial challenges, achieving a modest validation accuracy and witnessing signs of overfitting, as evidenced by a slight uptick in validation loss after the **12th epoch**.
- The introduction of the VGG16 model with initial layer freezing marked a significant improvement, registering a validation loss of **0.4122** and an accuracy of **84.60%** at the 20th epoch, demonstrating the effectiveness of leveraging pre-learned features.
- The subsequent unfreezing and fine-tuning of the VGG16 layers were even more consequential, leading to a marked validation loss reduction to **0.3776** and a leap in validation accuracy to **85.77%.**
- This fine-tuned VGG16 model, after meticulous hyperparameter tuning and strategic layer unfreezing, impressively achieved an accuracy of **92%**.
- This final accuracy figure not only attests to the model's enhanced predictive capabilities but also underscores the value of transfer learning coupled with fine-tuning in mastering the subtleties of your specific dataset, resulting in a robust and high-performing model.

<u>References:</u>

- **Research Papers**:
  - Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
  - Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 115(3), 211-252.
- **Books**:
  - "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville.
  - "Hands-On Machine Learning with Scikit-Learn, Kera's, and TensorFlow" by Aurelian Gerona.
- **Online Resources:**
- TensorFlow Hub: VGG16 Pre-trained Model
- https://www.tensorflow.org/api_docs/python/tf/keras/applications/vgg16/VGG16
- Kera's Applications: VGG16
  - https://keras.io/api/applications/vgg/#vgg16-function
  - **Additional Resources:**
- Image Classification with VGG16 Pre-trained Model
  - https://www.tensorflow.org/tutorials/images/classification