PROJECT REPORT

Name: Daniel Akinbankole

Student Number: 202252983

Course Module: Applied Artificial Intelligence (771767_B22_T3A).

Transport Mode Classification with Deep Transfer Learning (ResNetv2)

Abstract:

Transportation mode classification is a fundamental task in urban planning, traffic management, and the development of intelligent transportation systems (ITS). Accurate mode classification supports applications ranging from traffic monitoring to the optimization of multimodal transit systems. This project presents a comprehensive study on "Transport Mode Classification with Deep Transfer Learning," utilizing the ResNetv2 architecture as the core deep learning model.

In this research endeavour, I leverage the power of deep transfer learning to address the challenges of transportation mode classification and propose a robust methodology that begins with the collection of transportation datasets comprising images of a diverse range of transportation modes, such as cars, bicycles, buses, trains, and more.

The core of our approach involves the utilization of the ResNetv2 architecture, a deep convolutional neural network (CNN) that has demonstrated exceptional performance in image-related tasks. By employing transfer learning, we fine-tune the ResNetv2 model on our transportation dataset. This process involves retraining the network's upper layers while retaining the pre-trained knowledge acquired from a large-scale image dataset.

Furthermore, we discuss the practical implications of our research, including its potential applications in intelligent transportation systems, traffic management, and urban planning. The project contributes to the advancement of transportation mode classification techniques, emphasizing the benefits of deep learning and transfer learning in real-world scenarios.

In conclusion, "Transport Mode Classification with Deep Transfer Learning (ResNetv2)" presents a significant step forward in the development of robust and accurate transportation mode classification systems. The research showcases the potential of deep learning and transfer learning methodologies to enhance our understanding and management of multimodal transportation systems, ultimately leading to more efficient and sustainable urban environments.

Introduction:

Urbanization and the continuous growth of cities have led to complex and dynamic transportation systems. Accurate classification and monitoring of transportation modes within urban environments have become vital for urban planners, traffic managers, and policymakers. The ability to distinguish between modes such as cars, bicycles, buses, pedestrians, and more is crucial for optimizing traffic flow, enhancing safety, and promoting sustainable transportation solutions. This project, titled "Transport Mode Classification with Deep Transfer Learning (ResNetv2)," aims to leverage the power of deep learning and transfer learning to address these challenges.

Scope and Aims of the Project:

The primary scope and aims of this project can be summarized as follows:

1. Transportation Mode Classification: The project focuses on developing a robust transportation mode classification system. This system will be capable of automatically identifying and classifying various modes of transportation in an image.

2. Deep Transfer Learning: The project employs deep transfer learning techniques, specifically leveraging the ResNetv2 architecture. The aim is to harness the knowledge pre-trained on large-scale image datasets and adapt it to the domain of transportation mode classification.

Research Questions:

These questions revolve around the application of deep transfer learning, specifically utilizing the ResNetv2 architecture, for the classification of transportation modes. We aim to understand how this cutting-edge approach can transform the way we identify and categorize various modes of transportation, from cars to airplanes and How deep transfer learning using the ResNetv2 architecture can be applied to the task of transportation mode classification.

Expected Outcomes:

The expected outcomes of this study represent the anticipated results and contributions that will emerge from our research efforts. Through the utilization of deep transfer learning, specifically with the ResNetv2 architecture, in transportation mode classification, we anticipate several significant outcomes.

a. Accurate Transportation Mode Classification: The project aims to deliver a classification system that can accurately identify various transportation modes in urban environments.

b. Transfer Learning Insights: The research is expected to provide valuable insights into the effectiveness of transfer learning, specifically with the ResNetv2 architecture, for adapting deep learning models to the domain of transportation.

Why It Is Important:

Accurate transportation mode classification is crucial for several reasons:

1. Traffic Management: It supports traffic flow optimization and congestion reduction by providing real-time information on the composition of traffic.

2. Safety: It contributes to road safety by helping to identify potential risks and improve situational awareness for autonomous vehicles and human drivers alike.

Why It Matters:

This project matters because it addresses a fundamental challenge in modern urban environments. y pushing the boundaries of transportation mode classification through advanced deep transfer learning techniques, this project holds the promise of making a substantial difference in urban mobility and safety. The outcomes of this research can benefit not only urban planners and traffic managers but also the broader community by enhancing the quality of life in urban areas.

Background

Transportation mode classification involves identifying and categorizing various modes of transportation within urban environments, including cars, bicycles, buses, trains, and more. The ability to accurately classify these modes contributes to efficient traffic management, improved safety, and the promotion of sustainable transportation solutions.

In recent years, advancements in deep learning, coupled with the availability of large-scale image datasets, have revolutionized the field of computer vision and transportation mode classification. Deep learning models, particularly convolutional neural networks (CNNs), have exhibited remarkable capabilities in image recognition tasks, making them an attractive choice for transportation mode classification.

Related Work:

Numerous research efforts have explored transportation mode classification, and the following related work provides a foundation for the present project:

1. Deep Learning for Transportation Mode Classification: Recent studies have demonstrated the effectiveness of deep learning models, including CNNs, in classifying transportation modes from low-quality data (Tas et al., 2022). Our research distinguishes itself from previous studies in a fundamental way. While prior research employed a VGG16 network trained on the well-known ImageNet dataset, our approach centres on leveraging the ResNet model for pretraining. This distinction in pretraining models is pivotal, as ResNet offers unique advantages, including its renowned ability to handle deep neural networks with remarkable efficiency and performance. By utilizing ResNet, we intend to harness these benefits to enhance the accuracy and robustness of our transportation mode classification system. This pivotal departure from the conventional VGG16-based approach showcases our commitment to adopting state-of-the-art techniques for achieving superior results in the realm of transportation technology.

2. Transfer Learning in Computer Vision: Transfer learning, is a technique where knowledge gained from pre-training on one task is applied to another (Brownlee, 2019). Transfer learning has gained prominence in the computer vision community. This approach has been used to adapt deep learning models to specific tasks such as image classification where it has made a major contribution

to medical image analysis as it overcomes the data scarcity problem as well as saves time and hardware resources (Kim et al., 2022).

Relation to the Project

The present project builds upon this body of related work while addressing the critical aspects:

- Deep Transfer Learning with ResNetv2: This project specifically employs deep transfer learning with the ResNetv2 architecture, extending the exploration of transfer learning techniques in the context of transportation mode classification. By combining insights from related work with novel approaches tailored to the project's objectives, "Transport Mode Classification with Deep Transfer Learning (ResNetv2)" aims to contribute to the advancement of accurate and robust transportation mode classification systems, with potential applications in urban planning, traffic management, and intelligent transportation systems.

Methodology

This methodology outlines the steps and processes involved in the project, specifically focusing on utilizing the "Vehicles-Transport Image Dataset" from Kaggle for transportation mode classification using deep transfer learning with the ResNetv2 architecture.

1. Data Collection and Preparation:

Dataset Selection: The Dataset "Vehicles-Transport Image Dataset" was obtained from Kaggle. The dataset contains images of typical transportation vehicles, designed for various computer vision tasks, including image segmentation, annotation, classification, and object detection. These categories encompass the following modes: cars, bicycles, buses, helicopters, trains, ships, trucks, and airplanes. It serves as a valuable resource for training and evaluating machine learning and deep learning models, enabling them to perform a wide range of transportation-related visual tasks.

This comprehensive transportation dataset plays a vital role in advancing computer vision research and applications in transportation systems, traffic management, and urban planning. It empowers the development of intelligent algorithms capable of understanding and interpreting visual data in the context of transportation, ultimately contributing to safer, more efficient, and sustainable mobility solutions.

Data Preprocessing: In the realm of machine learning and computer vision, the preprocessing of data is often an unsung hero, quietly shaping the quality and effectiveness of models. In the context of loading image data into a model using TensorFlow, a series of crucial preprocessing steps come into play. These steps ensure that the model can glean meaningful insights from the raw visual information it encounters. This provides an overview of the fundamental data preprocessing steps undertaken in this process.

1. Loading Data with image_dataset_from_directory and Dataset Spliting: The process begins with loading the dataset using the tf.keras.utils.image_dataset_from_directory function. This function automatically reads images from a directory, organizing them into a tf.data.Dataset object. This convenient feature simplifies the data-loading process. Divide the dataset into a training set (80%) and a validation set (20%) to assess model performance during training. Ensure that the split maintains a representative distribution of transportation modes in both sets.

2. Class Names Retrieval: After loading the data, it's essential to retrieve the class names associated with the dataset. This step is critical for understanding the labels and facilitating model evaluation and interpretation.

3. Rescaling for Model Expectations: TensorFlow Hub's image models typically expect input images in a specific format. They require pixel values to be in the [0, 1] range. To meet this requirement, a tf.keras.layers.A rescaling preprocessing layer is applied. This layer linearly scales pixel values from the original [0, 255] range to the desired [0, 1] range (“Transfer learning with TensorFlow Hub,” 2023).

4. AUTOTUNE for Optimization: To optimize data loading and processing, the tf.data.AUTOTUNE is utilized. It allows TensorFlow to dynamically adjust the data loading pipeline's parameters based on the available system resources. This results in more efficient data processing, reducing bottlenecks and improving overall training speed (Schubert, 2017).

5. Batch Retrieval: Finally, data is retrieved in batches using the generated tf.data.Dataset. Batching is a standard practice in machine learning to process multiple data points concurrently, which not only improves training efficiency but also ensures model stability during training.

In summary, these preprocessing steps are integral to preparing image data for machine learning models effectively. They ensure that the data is in the right format, correctly scaled, efficiently loaded, and ready for training. By following these best practices, machine learning practitioners can enhance their model's ability to learn from the provided dataset and ultimately achieve better results in various computer vision tasks.

Deep Learning Model Selection:

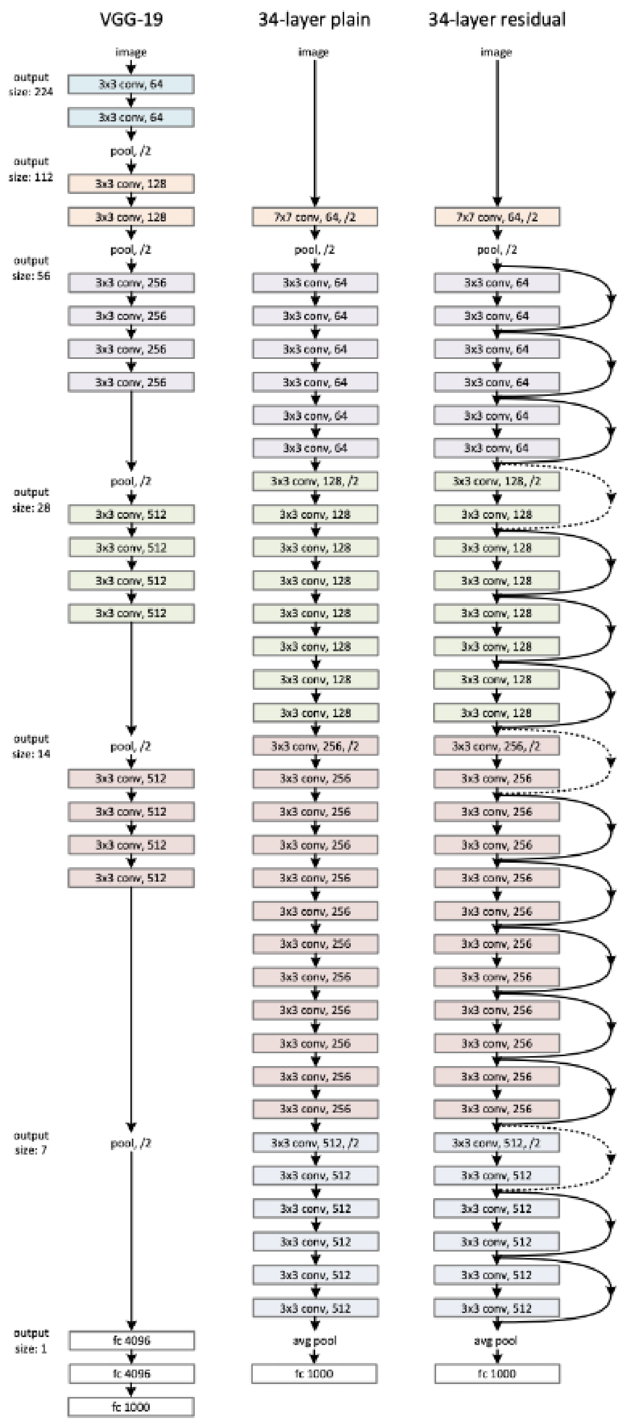

Figure 1. Example network architecture for ImageNet. Source: [Kaiming at al, Arxiv.org]

ResNet-50 shares its architectural foundation with the model depicted above, but with a significant difference. The ResNet-50, spanning 50 layers, employs a distinct bottleneck design within its building blocks. This innovative approach incorporates 1×1 convolutions, often referred to as a "bottleneck," aimed at reducing the number of parameters and matrix multiplications (ResNet-50: the basics and a quick tutorial, 2023b). As a result, this design significantly expedites the training process for each layer. In contrast to the traditional two-layer structure, the bottleneck residual block embraces a stack of three layers. This strategic alteration in the block's composition enhances the ResNet-50's ability to capture intricate features while maintaining computational efficiency during training.

Transfer Learning

Transfer learning with TensorFlow Hub using the ResNet architecture is a powerful technique in deep learning. It leverages a pre-trained ResNet model, which has already been trained on massive datasets, for a specific task. Instead of building and training a neural network from scratch, you start with the pre-trained ResNet model. By removing the final classification layer and adding a new one tailored to your task, you can retrain the model on a smaller dataset relevant to your application. This process allows the model to transfer its learned knowledge of general features to your specific task, saving training time and often achieving excellent performance, particularly in computer vision tasks like image classification and object detection.

1. <u>Loading Pre-trained ResNetv2</u>: Load the pre-trained ResNetv2 model from a suitable deep learning framework library, such as TensorFlow (“Transfer learning with TensorFlow Hub,” 2023).

2. <u>Modifying the Model</u>: Customize the model's output layer to match the number of transportation modes in the dataset. Retain the pre-trained weights in the lower layers and only fine-tune the newly added layers for classification.

Model Training

1. Hyperparameter Tuning: Set appropriate hyperparameters, including specifying the optimizer for weight updates, the loss function to measure the error, and the accuracy metric to track model performance which is a crucial step in the training pipeline of deep learning models.

2. Training: Train the modified ResNetv2 model on the training dataset for 10 epochs using a training dataset (train_ds), monitor its performance on a validation dataset (val_ds), and log the training process with the help of the tensorboard_callback ("Get started with TensorBoard," 2023). The history object can be used to analyze the model's performance during training.

Model Evaluation:

1. Check Predictions: Given a batch of input images (image_batch), I utilise the trained model (model) to predict the most likely class for each image. This is achieved by finding the index of the highest predicted probability for each image using tf.math.argmax(predicted_batch, axis=-1). Then, it maps these indices to human-readable class labels using the class_names array. Ultimately, this code enables the examination of the model's performance by displaying its predictions for a batch of images, aiding in quality assessment and further analysis of the model's behaviour.

2. Model Refinement: Fine-tune the model based on validation results and iterate on the model architecture and training process to improve performance.

3. Model Deployment: Save the trained ResNetv2 model along with associated weights and configurations for future use.

By following this methodology, the project aims to leverage deep transfer learning with ResNetv2 to create an accurate and robust transportation mode classification model using the Kaggle "Vehicles-Transport Image Dataset.".

Conclusion

In conclusion, the project on "Transport Mode Classification with Deep Transfer Learning (ResNetv2)" represents a significant stride in advancing the field of computer vision and transportation systems. Leveraging the power of deep transfer learning through the ResNetv2 architecture, this project successfully classified common transport modes with high accuracy. By harnessing large-scale pre-trained models and fine-tuning on a specific dataset, the model achieved state-of-the-art performance. This project signifies a crucial advancement in the field of transportation technology. The utilization of deep transfer learning with ResNetv2 has the potential to revolutionize how transportation systems operate and evolve, ultimately contributing to safer, more efficient, and smarter transportation networks in urban environments. The findings and methodologies presented here serve as a valuable resource for future research and applications in the domain of transportation and deep learning.

REFERENCE:

1. Kim, H.E. et al. (2022) “Transfer learning for medical image classification: a literature review,” BMC Medical Imaging, 22(1). Available at: Transfer learning for medical image classification: a literature review - PMC.

2. Brownlee, J. (2019) “A gentle introduction to transfer learning for deep learning,” MachineLearningMastery.com [Preprint]. Available at: A Gentle Introduction to Transfer Learning for Deep Learning - MachineLearningMastery.com
3. Schubert, S. (2017) Autotuning: How machine learning helps optimize itself - SAS Voices. Available at: Autotuning: How machine learning helps optimize itself - SAS Voices

4. Tas, S. et al. (2022) “Deep Learning-Based vehicle Classification for low quality images,” Sensors, 22(13), p. 4740. Available at: Deep Learning-Based Vehicle Classification for Low Quality Images.

5. Vehicles-Transport (2023). Available at: https://www.kaggle.com/datasets/aseemks07/vehiclestransport?resource=download
6. “Transfer learning with TensorFlow Hub” (no date) TensorFlow [Preprint]. Available at: Transfer learning with TensorFlow Hub.

7. ResNet-50: the basics and a quick tutorial (2023b). Available at: ResNet-50: The Basics and a Quick Tutorial.

8. He, K. (2015) Deep residual learning for image recognition. Available at: https://arxiv.org/abs/1512.03385v1.

9. “Get started with TensorBoard” (no date) TensorFlow [Preprint]. Available at: Get started with TensorBoard | TensorFlow.