Model Development Phase Template

Date	8 July 2024
Team ID	SWTID1720093035
Project Title	TechPart Vision: Personal Computer Parts Image Classification Using EfficientNet Transfer Learning
Maximum Marks	5 Marks

Model Selection Report

In the model selection report for future deep learning and computer vision projects, various architectures, such as CNNs or RNNs, will be evaluated. Factors such as performance, complexity, and computational requirements will be considered to determine the most suitable model for the task at hand.

Model Selection Report:

Model	Description
CNN using Sequential	A Convolutional Neural Network (CNN) implemented using Keras and TensorFlow for image classification of tech parts. The model consists of 2 layers including Convolutional layers, MaxPooling layers, Dropout layers, Flatten layer, and Dense layers. The activation function used in the convolutional and dense layers is ReLU , with softmax activation in the final dense layer for multi-class classification. The model is compiled with the Adam optimizer and categorical cross-entropy loss function, and it is trained for 10 epochs with a batch size of 32 with verbose 1 **its performance is very low and need a complex model as it underfitted**

EfficientNet V2B1	EfficientNetV2B1 was selected due to its balance of efficiency and performance, making it suitable for deployment scenarios with limited resources. The model leverages pre-trained weights on ImageNet, allowing for transfer learning which enhances accuracy with fewer training samples. The architecture was extended with custom layers to tailor it to our specific classification task. These layers include GlobalAveragePooling for reducing dimensions, BatchNormalization for stabilizing training, a Dense layer with ReLU activation and regularization to prevent overfitting, and a Dropout layer to further reduce overfitting risks. The output layer uses softmax activation to classify the input images into 14 different categories. **Despite its initial performance, we observed that EfficientNetV2B1 significantly outperformed other models like VGG19 in terms of training and validation accuracy and loss, which justified our decision to use it.**
VGG19	VGG19 is a deep convolutional neural network with 19 layers, pre-trained on the ImageNet dataset. For our project, we used VGG19 without the top classification layer (include_top=False) and added custom layers: a GlobalAveragePooling2D layer followed by two Dense layers (1024 units with ReLU activation and a final layer for classification with softmax activation). This setup leverages VGG19's powerful feature extraction capabilities. **Model Performance:** • **Training Accuracy and Loss: Showed significant improvement across epochs.** • **Validation Accuracy and Loss: Showed slower improvement and started to plateau, indicating potential overfitting (as seen in the provided graphs).** Rationale for Transitioning to EfficientNetV2B1: Based on the training and validation curves, VGG19 exhibited signs of overfitting, with the validation accuracy plateauing while training accuracy continued to improve. To address this and improve overall performance, we transitioned to EfficientNetV2B1.