

Model Optimization and Tuning Phase Template

Date	15 March 2024
Team ID	SWTID1720033149
Project Title	Visual Diagnostics: Detecting Tomato Plant Diseases With Leaf Image Analysis
Maximum Marks	10 Marks

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase involves refining neural network models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Hyperparameter Tuning Documentation (8 Marks):

Model	Tuned Hyperparameters
ResNet152V2	<p>Base Model: ResNet152V2,</p> <p>input_shape: (256, 256, 3),</p> <p>include_top: False,</p> <p>Frozen Layers: First 140 layers,</p> <p>Custom Layers: GlobalAveragePooling2D, Dense (1000 units, ReLU), Dense (10 units, softmax),</p> <p>Optimizer: Adam,</p> <p>Loss Function: Categorical Crossentropy, Metrics: Accuracy</p>

	<pre> from tensorflow.keras.applications import ResNet152V2 base_model = ResNet152V2(input_shape=(256,256,3), include_top=False) for layers in base_model.layers[:140]: layers.trainable=False for layers in base_model.layers[140:]: layers.trainable=True # Add your custom classification head x = GlobalAveragePooling2D()(base_model.output) x = Dense(1000, activation='relu')(x) # Adjust units as needed predictions = Dense(10, activation='softmax')(x) # Adjust number of classes accordingly # Create the final model model = Model(inputs=base_model.input, outputs=predictions) model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy']) </pre>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
ResNet152V	<p>Base Model: ResNet152V2 (pre-trained on ImageNet)</p> <p>Custom Layers: GlobalAveragePooling2D, Dense (1000 units with ReLU activation, followed by Dense with 10 units for softmax activation)</p> <p>Final Model: model created using Model(inputs=base_model.input, outputs=pred)</p> <p>The model incorporates the feature extraction capabilities of ResNet152V2 while fine-tuning its parameters to classify tomato plant diseases from leaf images effectively. This setup represents the final optimized model architecture tailored to specific project requirements.</p>

