

Model Optimization and Tuning Phase

Date	20 June 2025
Team ID	SWTID1749826875
Project Title	Dog Breed Identification using Transfer Learning
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
VGG19	<pre> from keras.callbacks import EarlyStopping from keras.optimizers import Adam opt = Adam(learning_rate=0.0001) # Define Early Stopping callback early_stopping = EarlyStopping(monitor='accuracy', patience=3, restore_best_weights=True) # Compile the model (you may have already done this) vgg19.compile(optimizer=opt , loss='categorical_crossentropy', metrics=['accuracy']) # Train the model with early stopping callback history = vgg19.fit(generator, epochs=50, callbacks=[early_stopping]) </pre> <ol style="list-style-type: none"> Batch Size = 32 Epochs = 50 Learning Rate = 0.0001

	<p>4. Optimizer: Adam optimizer</p> <p>5. Loss Function: categorical crossentropy</p> <p>6. Image Size = (224×224).</p>
MobileNetV2	<pre> from keras.callbacks import EarlyStopping from keras.optimizers import Adam opt = Adam(learning_rate=0.0001) # Define Early Stopping callback early_stopping = EarlyStopping(monitor='accuracy', patience=3, restore_best_weights=True) # Compile the model (you may have already done this) mn.compile(optimizer=opt , loss='categorical_crossentropy', metrics=['accuracy']) # Train the model with early stopping callback history = mn.fit(train_generator, epochs=10, callbacks=[early_stopping]) </pre> <p>1. Batch Size = 32</p> <p>2. Epochs = 10</p> <p>3. Learning Rate = 0.0001</p> <p>4. Optimizer: Adam optimizer</p> <p>5. Loss Function: categorical crossentropy</p> <p>6. Image Size = (224×224).</p>
EfficientNetB0	<pre> from keras.callbacks import EarlyStopping from keras.optimizers import Adam opt = Adam(learning_rate=0.0001) # Define Early Stopping callback early_stopping = EarlyStopping(monitor='accuracy', patience=3, restore_best_weights=True) # Compile the model (you may have already done this) en.compile(optimizer=opt , loss='categorical_crossentropy', metrics=['accuracy']) # Train the model with early stopping callback history = en.fit(train_generator, epochs=10, callbacks=[early_stopping]) </pre> <p>1. Batch Size = 32</p>

	<ol style="list-style-type: none"> 2. Epochs = 10 3. Learning Rate = 0.0001 4. Optimizer: Adam optimizer 5. Loss Function: categorical crossentropy 6. Image Size = (224×224).
--	--

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
MobileNetV2	<p>For this project, MobileNetV2 was chosen as the final model for dog breed classification using transfer learning. This decision was based on a combination of empirical performance and architectural advantages. Among the evaluated models, MobileNetV2 achieved the highest training and testing accuracy, demonstrating strong generalization capabilities even on a relatively small dataset of 1683 images across 20 classes. Its lightweight architecture, which uses depth-wise separable convolutions, allows it to maintain high efficiency while reducing the number of trainable parameters compared to larger models like VGG19. This made it particularly well-suited for our limited data scenario, where larger models tend to overfit. Additionally, MobileNetV2's pretrained weights on ImageNet allow it to transfer rich feature representations, significantly improving convergence speed and accuracy. The model is also highly compatible with deployment environments due to its low computational footprint, making it an ideal choice for real-time applications on mobile or web platforms.</p>