# Model Optimization and Tuning Phase Template

| | |
|---|---|
| Date | 15 June 2025 |
| Team ID | SWTID1750006853 |
| Project Title | ASL- Alphabet Image Recognition |
| 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 |
|---|---|
| Model 1: Custom CNN | **Learning Rate**: Adjusted from 0.01 to 0.001 for more stable convergence.<br><br>**Batch Size**: Experimented with 32, 64, and 128 — settled on 64 for memory-performance balance.<br><br>**Number of Conv Layers**: Tried 2–4 layers, found 3-layer configuration gave optimal balance of speed and performance.<br><br>`x = Conv2D(32, (3, 3), activation='relu', padding='same')(inputs)`<br>`x = MaxPooling2D((2, 2))(x)`<br><br>`x = Conv2D(64, (3, 3), activation='relu', padding='same')(x)`<br>`x = MaxPooling2D((2, 2))(x)`<br><br>`x = Conv2D(128, (3, 3), activation='relu', padding='same')(x)`<br>`x = MaxPooling2D((2, 2))(x)`<br><br>`x = Flatten()(x)`<br>`x = Dense(256, activation='relu')(x)`<br>`x = Dropout(0.5)(x)`<br>`outputs = Dense(29, activation='softmax')(x)`<br><br>`custom_cnn_model = Model(inputs, outputs)`<br><br>`custom_cnn_model.summary()` |

| | |
|---|---|
| Model 2: VGG16<br>(Transfer Learning) | **Learning Rate**: Tuned to 0.001 for the Adam optimizer.<br>**Dropout Rate**: Added 0.5 after dense layers to prevent overfitting.<br>**Trainable Layers**: All layers frozen initially; could optionally unfreeze some for fine-tuning.<br>`base_model = VGG16(weights='imagenet', include_top=False, input_shape=(CFG.img_height, CFG.img_width, CFG.img_channels))`<br>`for layer in base_model.layers:`<br>`  layer.trainable = False`<br>`x = base_model.output`<br>`x = Flatten()(x)`<br>`x = Dense(256, activation='relu')(x)`<br>`x = Dropout(0.5)(x)`<br>`x = Dense(512, activation='relu')(x)`<br>`x = Dropout(0.5)(x)`<br>`predictions = Dense(29, activation='softmax')(x)`<br>`model = Model(inputs=base_model.input, outputs=predictions)`<br>`display(model.summary())`<br>`display(tf.keras.utils.plot_model(model, to_file='vgg16.png', show_shapes=True))` |

**Final Model Selection Justification (2 Marks):**

| Final Model | Reasoning |
|---|---|
| VGG16 (Transfer Learning) | VGG16 was selected as the final model due to its superior classification performance on the ASL alphabet dataset. It achieved high accuracy on both training and validation sets, converged quickly due to pretrained weights, and benefited from strong feature extraction capabilities. Dropout layers reduced overfitting, and the model architecture was easily extendable for future enhancements (e.g., fine-tuning or more classes). Its performance exceeded that of a custom CNN and other lighter models like MobileNetV2, justifying its use as the production model. |