|  |
| --- |
| **Team ID :** LTVIP2025TMID32402 |
| Project Name : Enchanted Wings: Marvels of Butterfly Species |

**VGG16 Transfer-Learning Model**

**Overview**

The VGG16-based neural network in this project uses a pre-trained VGG16 model as a feature extractor. The original top (fully connected) layers are removed, and a custom classifier is added to adapt the model to the specific classification task.

**Key Components**

* Pre-trained Model: VGG16 (include\_top=False) loaded with ImageNet weights
* Input Shape: (224, 224, 3)
* Layer Freezing: All base layers are frozen
* Flatten Layer: Converts CNN output to 1D
* Dense Layer: SoftMax for classification into 28 categories

**Code Snippet**

python

CopyEdit

from tensorflow.keras.applications.vgg16 import VGG16

from tensorflow.keras.layers import Dense, Flatten

from tensorflow.keras.models import Model

vgg = VGG16(include\_top=False, input\_shape=(224, 224, 3))

for layer in vgg.layers:

layer.trainable = False

x = Flatten()(vgg.output)

output = Dense(28, activation='softmax')(x)

vgg16 = Model(vgg.input, output)

**Model Summary**

The model contains the VGG16 base layers followed by a Flatten layer and a Dense output layer.

**Architecture Snapshots**

Insert your screenshots here:

1. **VGG16 Layer Structure**
2. **Model Summary Output**

**Training**

* Optimizer: Adam
* Loss Function: Sparse Categorical Crossentropy
* Epochs: 15
* Callbacks: EarlyStopping, ModelCheckpoint
* Saved Model File: vgg16\_model.h5

### Model Building:

Vgg16 Transfer-Learning Model:

The VGG16-based neural network is created using a pre-trained VGG16 architecture with frozen weights. The model is built sequentially, incorporating the VGG16 base, a flattening layer, dropout for regularization, and a dense layer with SoftMax activation for classification into five categories. The model is compiled using the Adam optimizer and sparse categorical cross-entropy loss. During training, which spans 15 epochs, a generator is employed for the training data, and validation is conducted, incorporating call-backs such as Model Checkpoint and Early Stopping. The best-performing model is saved as "vgg16\_model.h5 " for potential future use. The model summary provides an overview of the architecture, showcasing the layers and parameters involved.



