Import Libraries:
The code begins by importing the required libraries: TensorFlow for deep learning operations, NumPy for numerical computations, and OpenCV for image processing.

Load Pretrained VGG19 Model:
A VGG19 model pretrained on the ImageNet dataset is loaded using TensorFlow's tf.keras.applications.VGG19 class. This model will be used to extract features from the content and style images.

Define Content and Style Layers:
Lists of layers are defined to specify which layers of the VGG19 model will be used to capture content and style information. These layers' outputs will be used to compute the content and style losses.

Preprocess Image:
The preprocess_image function takes an image path as input, reads and converts the image to the appropriate format, resizes it to a specified size, and normalizes pixel values to the range [0, 1].

Compute Gram Matrix:
The gram_matrix function calculates the Gram matrix of a given tensor. The Gram matrix is used to represent the correlations between feature maps and is a key component in computing style loss.

Create Content and Style Models:
The get_content_model and get_style_model functions create models that take an image as input and produce the outputs of specific layers defined earlier. These models will be used to extract content and style features.

Clip Pixel Values:
The clip_0_1 function ensures that pixel values of an image are constrained within the range [0, 1].

Perform Style Transfer:
The style_transfer function is the heart of the style transfer process. It takes a content image and a style image along with optional parameters. It preprocesses the images, computes content and style features using the models created earlier, initializes a generated image, and then iteratively updates the generated image using gradient descent to minimize a combination of content and style losses.

Main Code:
The main part of the code specifies paths to the content and style images, performs the style transfer process, and then post-processes the output image to display, save, and show it.

Terminologies:

VGG19 Model: A deep convolutional neural network architecture used for image classification and feature extraction.

Content and Style Layers: Specific layers in the VGG19 model that capture content and style information from images.

Gram Matrix: A matrix that represents the correlations between feature maps of a neural network layer.

Content Loss: A measure of the difference between the content features of the generated image and the content image.

Style Loss: A measure of the difference between the style features of the generated image and the style image.

Optimizer: An algorithm that adjusts the generated image to minimize the defined loss function.

Gradient Descent: An optimization technique that iteratively adjusts the parameters (in this case, pixel values of the generated image) to minimize the loss.

Total Loss: A combination of content and style losses, weighted by content and style weights, used to guide the optimization process.