# Image Histogram and HOG Feature Visualization Documentation

This document describes the functions and steps used to visualize histograms of pixel intensities and Histogram of Oriented Gradients (HOG)   
for feature extraction from images. The code performs several tasks, including reading, resizing, and analyzing the image using HOG and pixel intensity histograms.

## Modules Required:

- numpy  
- cv2 (OpenCV)  
- matplotlib.pyplot  
- skimage.feature.hog  
- skimage.exposure

## Steps:

### Step 1: Read the Image

Reads the input image in grayscale format using OpenCV. Optionally, use `cv2.IMREAD\_COLOR` for color images.

### Step 2: Resize Image

Resizes the image to 224x224 pixels for uniformity in processing. This step ensures that all images have the same dimensions,   
which is useful when applying certain filters or analyses that depend on image size.

### Step 3: Plot Basic Histogram

The `plot\_basic\_histogram(image)` function flattens the pixel values into a 1D array and creates a histogram showing pixel intensity frequency.  
This plot is useful for understanding image brightness and contrast distribution.

### Step 4: Compute and Plot HOG

The `plot\_original\_and\_hog(image)` function computes the Histogram of Oriented Gradients (HOG) features for edge detection and visualizes   
the HOG image alongside the original. The function provides insight into the prominent edges and features within an image.

### Step 5: Plot Resized Image and Histogram Side by Side

The `plot\_image\_and\_histogram(image)` function displays both the resized image and its histogram in a single figure. This combined view provides   
both the visual structure of the image and its pixel intensity distribution, aiding in quick visual analysis.

## Usage Instructions:

1. Update `image\_path` to the correct path of the image you want to analyze.  
2. Ensure all necessary libraries are installed and imported.  
3. Run the code to visualize the image, pixel intensity histogram, and HOG feature map.