Shirke Aryan 21BCS111

```
#include <opencv2/opencv.hpp>
#include <iostream>

cv::Mat customResize(const cv::Mat& input, double scale_x,
double scale_y, int interpolation) {
cv::Mat output;
cv::resize(input, output, cv::Size(), scale_x, scale_y,
interpolation);
return output;
}

int main() {
    Load images
cv::Mat image1 = cv::imread("IttigatiScene.jpg");
cv::Mat image2 = cv::imread("BS-1Pedha.jpg");
cv::Mat image3 = cv::imread("lena_gray.bmp");

    Test cases
std::vector> testCases = {
{image1, 0.5, 0.7, cv::INTER_NEAREST},
{image1, 0.75, 0.7, cv::INTER_LINEAR},
{image2, 400.0  2448.0, 300.0  1770.0,
cv::INTER_CUBIC},
```

```
{image3, 1280.0 / 512.0, 720.0 / 512.0,
cv::INTER_NEAREST},
{image3, 1920.0 / 512.0, 1080.0 / 512.0,
cv::INTER_LINEAR}
    };

    // Compare custom resize with OpenCV resize
for (int i = 0; i < testCases.size(); ++i) {
cv::Mat input, resized_custom, resized_opencv;
double scale_x, scale_y;
int interpolation;

std::tie(input, scale_x, scale_y, interpolation) =
testCases[i];

    // Custom resize
resized_custom = customResize(input, scale_x, scale_y,
interpolation);

    // OpenCV resize
cv::resize(input, resized_opencv, cv::Size(), scale_x,
scale_y, interpolation);

    // Calculate point-wise difference
cv::Mat diff;
```

```
cv::absdiff(resized_custom, resized_opencv, diff);

    Display results
cv::imshow("Original", input);
cv::imshow("Custom Resize", resized_custom);
cv::imshow("OpenCV Resize", resized_opencv);
cv::imshow("Point-wise Difference", diff);
cv::waitKey(0);
    }

return 0;
}
```

This C++ program defines a customResize function that performs image resizing with different interpolation techniques. The test cases involve resizing images with different scales and interpolation methods. The program compares the results of custom resizing and OpenCV's resizing by calculating the point-wise difference and displays the original image, custom resized image, OpenCV resized image, and point-wise difference for each test case.

Make sure to replace the image file names with the actual file names and adjust the test cases accordingly. Also, note that the error calculation can be further refined based on