//képbetöltés

Mat img = imread("kep.png", IMREAD\_GRAYSCALE);

// Create one images with the same size as img, in 16-bit signed format   
Mat G1;  
 img.convertTo(G1, CV\_16S);

// Compute G1(x,y) = I(x,y) - I(x+1, y+1)

for (int y = 0; y < img.rows - 1; y++) { for (int x = 0; x < img.cols - 1; x++) { G1.at<short>(y, x) = img.at<uchar>(y, x) - img.at<uchar>(y + 1, x + 1); } }

// Approximate gradient strength:

G = |G1| + |G2| Mat G = abs(G1) + abs(G2); // Convert G to 8-bit for display Mat G8U; convertScaleAbs(G, G8U);

// Display gradient magnitude image

imshow("Gradient Magnitude", G8U); // Threshold to suppress false edges Mat edgeImg; double threshold\_value = 100;

// Adjust this value as needed

threshold(G8U, edgeImg, threshold\_value, 255, THRESH\_BINARY);

// Display the final edge-detected image

imshow("Edge Image", edgeImg);

waitKey(0);

return 0;