import numpy as np

import cv2

# Read the query image as query\_img

# and train image This query image

# is what you need to find in train image

# Save it in the same directory

# with the name image.jpg

query\_img = cv2.imread('query.jpg')

train\_img = cv2.imread('train.jpg')

# Convert it to grayscale

query\_img\_bw = cv2.cvtColor(query\_img,cv2.COLOR\_BGR2GRAY)

train\_img\_bw = cv2.cvtColor(train\_img, cv2.COLOR\_BGR2GRAY)

# Initialize the ORB detector algorithm

orb = cv2.ORB\_create()

# Now detect the keypoints and compute

# the descriptors for the query image

# and train image

queryKeypoints, queryDescriptors = orb.detectAndCompute(query\_img\_bw,None)

trainKeypoints, trainDescriptors = orb.detectAndCompute(train\_img\_bw,None)

# Initialize the Matcher for matching

# the keypoints and then match the

# keypoints

matcher = cv2.BFMatcher()

matches = matcher.match(queryDescriptors,trainDescriptors)

# draw the matches to the final image

# containing both the images the drawMatches()

# function takes both images and keypoints

# and outputs the matched query image with

# its train image

final\_img = cv2.drawMatches(query\_img, queryKeypoints,

train\_img, trainKeypoints, matches[:20],None)

final\_img = cv2.resize(final\_img, (1000,650))

# Show the final image

cv2.imshow("Matches", final\_img)

cv2.waitKey(3000)