

A. Identitas

NIM : 2103065

Nama Lengkap : Deo Ananda Rizky

Kelas : D3TI2C

Program Studi : D3 TEKNIK INFORMATIKA

Jurusan : TEKNIK INFORMATIKA

PENJELASAN CODE PROGRAM

Penjelasan Program Nomor 1

import numpy as np - Mengimpor library numpy dan memberikan alias np.

import cv2 as cv - Mengimpor library OpenCV dan memberikan alias cv.

filename = 'catur.jpeg' - Mendefinisikan variabel filename dengan nama file gambar yang akan diproses.

img = cv.imread(filename) - Membaca gambar dari file yang ditentukan di filename menggunakan fungsi imread dari OpenCV.

gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY) - Mengubah gambar warna img menjadi grayscale menggunakan fungsi cvtColor dari OpenCV.

gray = np.float32(gray) - Mengubah gambar grayscale gray ke format floating point menggunakan fungsi float32 dari numpy.

dst = cv.cornerHarris(gray, 2, 3, 0.04) - Mendeteksi sudut-sudut pada gambar menggunakan algoritma Harris Corner Detection dan fungsi cornerHarris dari OpenCV. Parameter 2, 3, dan 0.04 adalah ukuran tetangga yang dipertimbangkan untuk deteksi sudut, ukuran kernel Sobel yang digunakan untuk menghitung turunan, dan parameter bebas detektor Harris, secara berturut-turut.

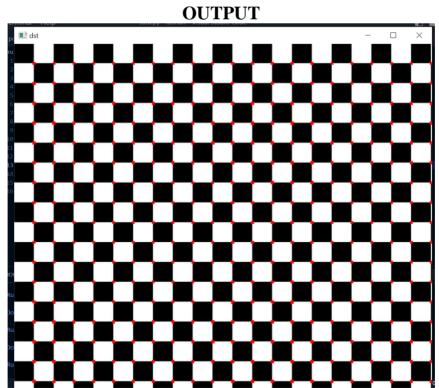
dst = cv.dilate(dst, None) - Membesarkan titik sudut pada gambar menggunakan fungsi dilate dari OpenCV.

img[dst > 0.01 * dst.max()] = [0, 0, 255] - Menandai sudut-sudut yang terdeteksi pada gambar dengan mengubah warna piksel menjadi merah. Sudut-sudut dideteksi berdasarkan ambang 0.01 kali nilai maksimum dari dst.

cv.imshow('dst', img) - Menampilkan gambar akhir dengan sudut-sudut yang terdeteksi menggunakan fungsi imshow dari OpenCV.

if cv.waitKey(0) & 0xff == 27: - Menunggu suatu event keyboard dan memeriksa apakah itu adalah tombol 'Esc' (kode ASCII 27).

cv.destroyAllWindows() - Menutup semua jendela OpenCV yang terbuka selama eksekusi program.



Penjelasan Program Nomor 2

import numpy as np

import cv2 as cv -> Kode di atas mengimpor dua modul yaitu NumPy dan OpenCV.

filename = 'catur.jpeg'

img = cv.imread(filename) -> membaca gambar 'chessboard2.jpg' dan menyimpannya dalam variabel img.

gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)->mengonversi gambar berwarna menjadi gambar grayscale (grayscale adalah gambar dengan tingkat keabuan atau hanya memiliki satu channel warna). Konversi ini dilakukan dengan menggunakan fungsi cvtColor() dari modul OpenCV. Dalam kasus ini, cvtColor() digunakan untuk mengubah gambar img dari mode warna BGR (Blue, Green, Red) menjadi mode grayscale.

gray = np.float32(gray)->mengubah tipe data dari gambar grayscale menjadi tipe data float32 menggunakan fungsi float32() dari NumPy.

dst = cv.cornerHarris(gray, 2, 3, 0.04) -> menghitung sudut-sudut sudut Harris pada gambar grayscale menggunakan fungsi cornerHarris() dari OpenCV. Parameter pertama gray adalah gambar grayscale yang telah dikonversi sebelumnya, parameter kedua dan ketiga adalah ukuran kernel sobel untuk menghitung gradien gambar, dan parameter terakhir adalah konstanta sudut Harris

dst = cv.dilate(dst, None) -> melakukan operasi dilasi pada gambar sudut Harris menggunakan fungsi dilate() dari OpenCV. Tujuan dilasinya gambar adalah untuk memperkuat nilai sudut pada gambar.

ret, dst = cv.threshold(dst, 0.01*dst.max(), 255, 0) -> melakukan thresholding pada gambar sudut Harris dengan nilai ambang sebesar 1% dari nilai maksimum sudut Harris menggunakan fungsi threshold() dari OpenCV.

dst = np.uint8(dst) ->mengubah tipe data gambar sudut Harris menjadi tipe data uint8 (8-bit unsigned integer) menggunakan fungsi uint8() dari NumPy.

ret, labels, stats, centroids = cv.connectedComponentsWithStats(dst) ->menghitung komponen-komponen terhubung dari gambar sudut Harris menggunakan fungsi connectedComponentsWithStats() dari OpenCV. Hasil dari fungsi ini adalah jumlah komponen terhubung, label-label komponen, statistik statistik komponen, dan koordinat-kordinat centroid dari setiap komponen.

criteria = (cv.TERM_CRITERIA_EPS + cv.TERM_CRITERIA_MAX_ITER, 100, 0.001) corners = cv.cornerSubPix(gray, np.float32(centroids), (5,5), (-1,-1), criteria) ->melakukan sub-pixel corner refinement pada setiap sudut sudut sudut Harris menggunakan fungsi cornerSubPix() dari OpenCV. Fungsi ini mengambil gambar grayscale, posisi-posisi sudut Harris yang telah dihitung sebelumnya, ukuran kernel sobel, posisi for i in range(len(corners)):

x, y = corners[i]

cv.circle(img, (int(x), int(y)), 5, (0, 0, 255), -1) ->Menggambar titik sudut pada gambar menggunakan fungsi cv.circle() dari OpenCV dan mengubah warna titik sudut menjadi merah.

cv.imshow('corners', img)

if cv.waitKey(0) & 0xff == 27:

cv.destroyAllWindows() ->Menampilkan gambar hasil pada jendela pop-up menggunakan fungsi cv.imshow() dari OpenCV dan menunggu input keyboard. Jika tombol escape (kode ASCII 27) ditekan, maka jendela akan ditutup menggunakan fungsi cv.destroyAllWindows() dari OpenCV.

OUTPUT

