15.

import numpy as np

import cv2

import matplotlib.pyplot as plt

from google.colab import files

uploaded = files.upload()

img\_path = next(iter(uploaded))

img = cv2.imread(img\_path)

img = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)

src\_pts = np.array([

[100, 100],

[400, 100],

[400, 400],

[100, 400]

], dtype=np.float32)

dst\_pts = np.array([

[0, 0],

[300, 0],

[300, 300],

[0, 300]

], dtype=np.float32)

def compute\_homography\_dlt(src, dst):

num\_points = src.shape[0]

A = []

for i in range(num\_points):

x, y = src[i][0], src[i][1]

u, v = dst[i][0], dst[i][1]

A.append([-x, -y, -1, 0, 0, 0, x\*u, y\*u, u])

A.append([ 0, 0, 0, -x, -y, -1, x\*v, y\*v, v])

A = np.array(A)

Vt = np.linalg.svd(A)

H = Vt[-1].reshape(3, 3)

return H / H[2, 2]

H\_dlt = compute\_homography\_dlt(src\_pts, dst\_pts)

output\_size = (300, 300)

warped\_img = cv2.warpPerspective(img, H\_dlt, output\_size)

plt.figure(figsize=(10, 5))

plt.subplot(1, 2, 1)

plt.title("Original Image")

plt.imshow(img)

plt.scatter(src\_pts[:, 0], src\_pts[:, 1], c='red')

plt.subplot(1, 2, 2)

plt.title("Transformed Image (DLT)")

plt.imshow(warped\_img)

plt.show()

OUTPUT:

