第三次作业

一、主要功能

有重叠区域图像拼接，绘制匹配点图、单应性转换图、拼接效果图。

二、代码实现

# Work 2

# 拼接有重叠区域图像

def stitch(img\_A, img\_B):

    sift = cv.SIFT\_create()

    kp1, des1 = sift.detectAndCompute(img\_A, None)

    kp2, des2 = sift.detectAndCompute(img\_B, None)

    bf = cv.BFMatcher()

    matches = bf.knnMatch(des1, des2, k=2)

    good\_matches = []

    for m, n in matches:

        if m.distance < 0.7 \* n.distance:

            good\_matches.append(m)

    src\_pts = np.float32([kp1[m.queryIdx].pt for m in good\_matches]).reshape(-1, 1, 2)

    dst\_pts = np.float32([kp2[m.trainIdx].pt for m in good\_matches]).reshape(-1, 1, 2)

    M, \_ = cv.findHomography(src\_pts, dst\_pts, cv.RANSAC, 5.0)

    dst = cv.warpPerspective(img\_A, M, (img\_A.shape[1] + img\_B.shape[1], img\_A.shape[0]))

    dst[0:img\_B.shape[0], 0:img\_B.shape[1]] = img\_B

    return dst, draw\_matches(img\_A, img\_B, kp1, kp2, good\_matches), draw\_homography(img\_A, img\_B, kp1, kp2, good\_matches)

# 绘制匹配点

def draw\_matches(img\_A, img\_B, kp1, kp2, good\_matches):

    img\_A = cv.cvtColor(img\_A, cv.COLOR\_BGR2RGB)

    img\_B = cv.cvtColor(img\_B, cv.COLOR\_BGR2RGB)

    img3 = cv.drawMatches(img\_A, kp1, img\_B, kp2, good\_matches, None, flags=2)

    # plt.figure()

    # plt.imshow(img3)

    # plt.axis('off')

    # plt.show()

    return img3

# 绘制单应性转换图

def draw\_homography(img\_A, img\_B, kp1, kp2, good\_matches):

    img\_A = cv.cvtColor(img\_A, cv.COLOR\_BGR2RGB)

    img\_B = cv.cvtColor(img\_B, cv.COLOR\_BGR2RGB)

    src\_pts = np.float32([kp1[m.queryIdx].pt for m in good\_matches]).reshape(-1, 1, 2)

    dst\_pts = np.float32([kp2[m.trainIdx].pt for m in good\_matches]).reshape(-1, 1, 2)

    M, \_ = cv.findHomography(src\_pts, dst\_pts, cv.RANSAC, 5.0)

    h, w = img\_A.shape[:2]

    pts = np.float32([[0, 0], [0, h - 1], [w - 1, h - 1], [w - 1, 0]]).reshape(-1, 1, 2)

    dst = cv.perspectiveTransform(pts, M)

    img2 = cv.polylines(img\_B, [np.int32(dst)], True, 255, 3, cv.LINE\_AA)

    # plt.figure()

    # plt.subplot(121)

    # plt.imshow(img\_A)

    # plt.title('Input\_A')

    # plt.axis('off')

    # plt.subplot(122)

    # plt.imshow(img2)

    # plt.title('Output\_A')

    # plt.axis('off')

    # plt.show()

    return img2

def main\_2():

    img\_A = cv.imread('./0417dataset/images/DSC02932.JPG')

    img\_B = cv.imread('./0417dataset/images/DSC02931.JPG')

    dst, matches, img2 = stitch(img\_A, img\_B)

    plt.figure()

    plt.imshow(dst[:,:,::-1])

    plt.axis('off')

    plt.figure()

    plt.imshow(matches)

    plt.axis('off')

    plt.figure()

    plt.subplot(121)

    plt.imshow(img\_A[:,:,::-1])

    plt.title('Input\_A')

    plt.axis('off')

    plt.subplot(122)

    plt.imshow(img2)

    plt.title('Output\_A')

    plt.axis('off')

main\_2()

三、运行结果





