CVPR Homework 1

——王申奥 人工智能学院 3122155013

1. 图像的 SIFT 特征检测原理

1.1 SIFT 简介

SIFT，即尺度不变特征变换（Scale-invariant feature transform，SIFT），是用于图像处理领域的一种描述。这种描述具有尺度不变性，可在图像中检测出关键点，是一种局部特征描述子。该方法于 1999 年由 David Lowe 首先发表于计算机视觉国际会议(International Conference on Computer Vision，ICCV)[1]，2004 年再次经 David Lowe 整理完善后发表于 International journal of computer vision(IJCV)[2]。

引入 SIFT 特征是为了解决图片的匹配问题，即想要从图像中提取一种对图像的大小和旋转变化保持鲁棒的特征，从而实现匹配。这一算法的灵感也十分直观：人眼观测两张图片是否匹配时会注意到其中的典型区域（特征点部分）。如果我们能够实现这一特征点区域提取过程，再对所提取到的区域进行描述，便可在此基础上实现特征匹配。于是特征提取问题就演变成了以下几个子问题：

a) 应该选取什么样的点作为特征点呢？
b) 怎样使得选取的特征点有尺度不变性呢？
c) 怎样使得选取的特征点有缩放不变性呢？
d) 怎样使得选取的特征点有旋转不变性呢？
e) 如何描述特征点区域呢？

针对子问题 a：由于人眼对图像中的高频区域更加的敏感，因此我们应该选择变化剧烈的边缘或者角点进行检测；同时我们选择的对象是特征点，而边缘往往由连续的一些点组成，所以这里我们选择检测角点。针对子问题 b，c，d：我们引入高斯金字塔来获取原始图像在不同尺度下的图像变体，由这些图像变体获得尺度不变特征点；再缩放到原始图像尺度来得到具有缩放不变性的特征点；后续采取将特征点区域旋转到主方向的设定来保证特征点的旋转不变性。针对子问题 e：我们使用特征点区域内每个方向的梯度赋值来描述特征区域。

1.2 SIFT 特征检测算法

SIFT 特征对图像尺度和旋转是不变的，并且对仿射失真、三维变化、噪声和光照变化具有很强的鲁棒性。下面介绍如何从原始图像中获取 SIFT 特征，即 SIFT 特征检测算法的算法流程。SIFT 特征检测算法包含以下四部分：生成尺度空间、关键点定位、关键点方向和关键点描述。

1.2.1 生成尺度空间

算法第一阶段是找到对图像尺度变化不变的图像位置。这是通过在所有可能的尺度上使用一个称为尺度空间的函数搜索稳定的特征实现的。尺度空间将图像表示为平滑后的图像的一个参数簇，目的是模拟图像的尺度减小时出现的细节损失。控制平滑的参数称为尺度参数。

Lindeberg 指出：在尺度不变性假设下，高斯核及其导数是尺度空间分析中唯一可能的平滑核[3]。因此在 SIFT 中，使用高斯核进行平滑，尺度参数为标准差。

定义一张灰度图像 $f(x,y)$ 的尺度空间 $L(x,y,\sigma)$ 为：

$$L(x,y,\sigma)=G(x,y,\sigma)\otimes f(x,y)\text{，}\otimes\text{表示空间卷积} \tag{1-1}$$

式中：σ ——尺度参数，$G(x,y,\sigma)=\frac{1}{2\pi\sigma^2}e^{\frac{-(x^2+y^2)}{2\sigma^2}}$ 。

输入图像 $f(x,y)$ 依次与标准差为 $\sigma,k\sigma,k^2\sigma,k^3\sigma,\cdots$ 的高斯核进行空间卷积，生成一堆由常量因子 k 分割的高斯平滑图像，如图 1-1 所示。

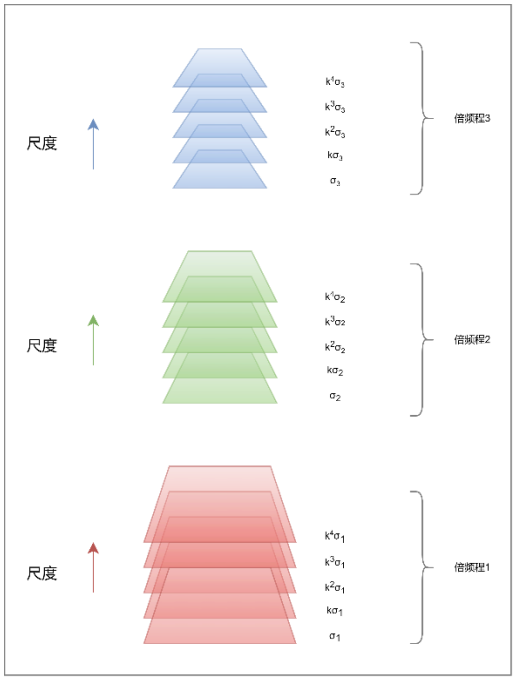


图 1-1 尺度空间示意图，显示了 3 倍频程

1.2.2 关键点定位

SIFT 最初使用高斯滤波后的图像来查找关键点的位置，然后使用两个处理步骤来改善这些关键点的位置和正确性。

a) 查找初始关键点

$$D(x,y,k\sigma)=[G(x,y,(k+1)\sigma)-G(x,y,k\sigma)]\otimes f(x,y) \tag{1-2}$$

SIFT 检测首先检测一个倍频程中两幅相邻尺度空间图像的高斯差的极值与原始图像的卷积来确定关键点的位置，即查找上式的极值。如图 1-2 所示。

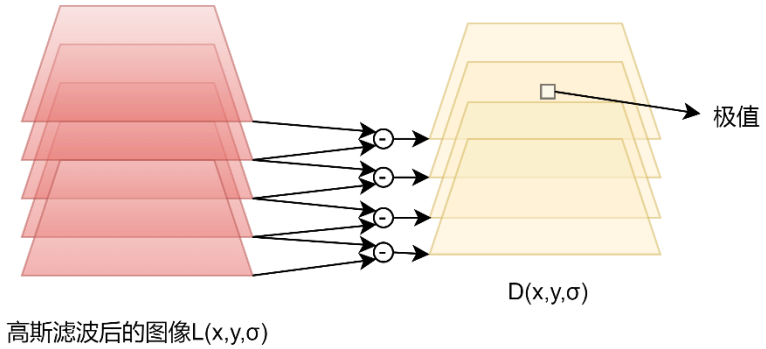


图 1-2 查找初始关键点示意图

b) 改进关键点位置的精度

由于一个函数被抽样时，它真正的最大值或最小值实际上位于样本点之间。得到真实极值（亚像素精度）的一种方法是在数字函数的每个极值点处拟合一个内插函数，然后在内插后的函数中查找改进精度的极值位置。

SIFT 用 $D(x,y,\sigma)$ 的泰勒级数展开的线性项和二次项，把原点移至被检测的关键点，

即下式：

$$D(x) = D + (\nabla D)^{\mathrm{T}} x + \frac{1}{2} x^{\mathrm{T}} H x \tag{1-3}$$

取上式中 x 的导数为 0，可得到极值的精准位置 $\hat{x}$ 为：

$$\hat{x} = -H^{-1}(\nabla D) \tag{1-4}$$

SIFT 使用极值位置的函数值来剔除具有低对比度的不稳定极值。将式(1-4)代入(1-3)可得:

$$D(\hat{x}) = D + \frac{1}{2}(\nabla D)^{\mathrm{T}} \hat{x} \tag{1-5}$$

当 $D(\hat{x}) < threshold$ 时，该关键点则会被剔除。

c) 消除边缘响应

上述原始图像处理过程得到的关键点包括图像中的边缘，但 SIFT 中关键点是更加局部的“角点”，因此需要剔除初始关键点中边缘部分。由于边在一个方向上具备高曲折度，在正交方向上具有低曲折度，因此引入图像中局部曲折度。

图像中某点的曲折度可由该点出的海森矩阵算出，即：

$$H = \begin{pmatrix} D_{xx} & D_{xy} \\ D_{yx} & D_{yy} \end{pmatrix} \tag{1-6}$$

令：

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta$$
$$Det(H) = D_{xx} D_{yy} - D_{xy}^2 = \alpha\beta$$
$$\alpha = r\beta$$

式中 α 和 β 分别为 H 的最大特征值和最小特征值， r 为两者之比。当：

$$\frac{[Tr(H)]^2}{Det(H)} < \frac{(r+1)^2}{r} \tag{1-7}$$

则剔除该关键点。通过这一步操作可以剔除边缘关键点。

1.2.3 关键点方向

通过上述步骤，我们得到了每个稳定的关键点在尺度空间中的位置，实现了尺度独立性，在这一步。我们根据图像的局部性质为每个关键点分配一个一致的方向，以便可以表示相对于其方向的关键点，实现图像的旋转不变性。

使用关键点的尺度来选择最接近该尺度的高斯平滑图像 $L(x, y)$，得到关键点处的像素差，根据式(1-8)、(1-9 计算梯度幅度 $M(x, y)$ 和方向角 $\theta(x, y)$。

$$M(x, y) = [(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2]^{\frac{1}{2}} \tag{1-8}$$

$$\theta(x, y) = \arctan[(L(x, \mathrm{y}+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y))] \tag{1-9}$$

在完成关键点的梯度计算后，使用直方图统计邻域内像素的梯度和方向。梯度直方图有 36 个容器，覆盖了图像平面上 360° 的方向范围。添加到直方图中的每个样本按期梯度幅度

直方图的各个峰值对应于局部梯度的主要局部方向。在直方图中检测到的最高峰值及这个最高峰值 80%的其他峰值也用于创建具有这一方向的关键点。因此，对于具有类似幅度的多个峰值的位置，这将在相同的位置上以相同的幅度创建多个关键点，但这些关键点的方向不同。SIFT 仅为 15%左右的多方向点分配多个方向，的那他们对图像匹配的贡献非常明显。

1.2.4 关键点描述

通过以上步骤，对于每一个关键点，拥有三个信息：位置、尺度以及方向。接下来就是为每个关键点建立一个描述符，用一组向量将这个关键点描述出来，使其不随各种变化而改变，比如光照变化、视角变化等等。这个描述子不但包括关键点，也包含关键点周围对其有贡献的像素点，并且描述符应该有较高的独特性，以便于提高特征点正确匹配的概率。SIFT 描述子是关键点邻域高斯图像梯度统计结果的一种表示。通过对关键点周围图像区域分块，计算块内梯度直方图，生成具有独特性的向量，这个向量是该区域图像信息的一种抽象，具有唯一性。

下面介绍如何生成特征点区域描述子：首先在特征点区域所处的尺度层面划定特征区域，半径 $r=\frac{3\sqrt{2}\sigma_{oct}(d+1)}{2}$，其中 d 为我们在一个维度上划分的子块数目，通常为 4。将该区域划分为共 $d\times d$ 个子块，每个子块内包含多个像素点。将规划出来的区域旋转到该区域的主方向，以保证旋转不变性。在每个子块内构筑幅值-方向直方图，统计 8 各方向的梯度幅值。由此在该区域可用 $8\times d\times d$ 维向量表示。通过将该向量的每一分量除以向量范数做归一化处理，降低光照对描述子的影响。

上述过程如图 1-3 所示：

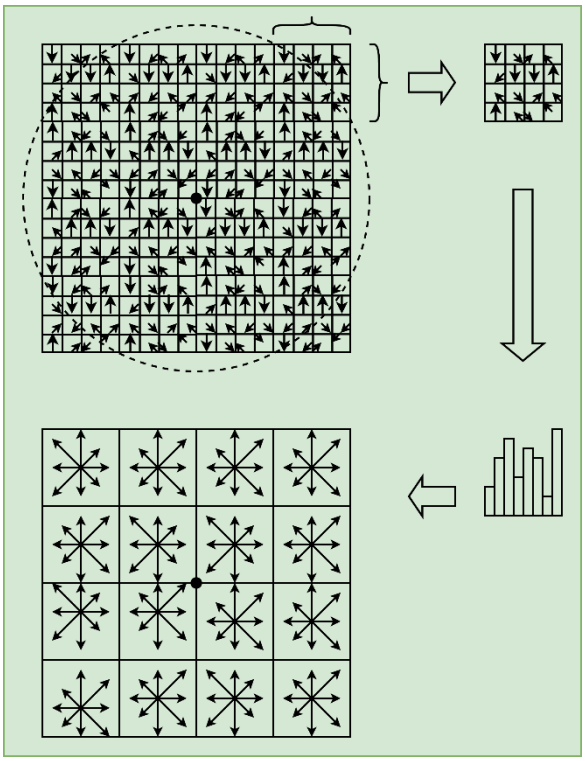


图 1-3, SIFT 描述子生成过程示意图

1.3 小结

SIFT 特征是图像的局部特征，其对旋转、尺度缩放、亮度变化保持不变性，对视角变化、仿射变换、噪声也保持一定程度的稳定性； 但对边缘光滑的目标无法准确提取特征点。

2. 图像特征描述子的匹配度量（距离比），RANSAC 方法

2.1 距离比

关键点的匹配可以采用穷举法来完成，但是这样耗费的时间太多，一般都采用 kd 树的数据结构来完成搜索。搜索的内容是以目标图像的关键点为基准，搜索与目标图像的特征点 R_i 最邻近的原图像特征点 S_j 和次邻近的原图像特征点 S_k。

定义模板图中的关键点描述子：$R_i=(r_{i1},r_{i2},\cdots,r_{i128})$；定义匹配图中关键点的描述子：$S_i=(s_{i1},s_{i2},\cdots,s_{i128})$；定义任意两描述子相似性度量为描述子向量距离：$d(R_i,S_i)=\sqrt{\sum_{j=1}^{128}\left(r_{ij}-s_{ij}\right)^2}$。则得到配对的关键点描述子需满足：

$$\frac{d(R_i,S_j)}{d\left(R_i,S_k\right)}<threshold \tag{2-1}$$

上式即图像特征匹配的匹配度量距离比。

2.2 RANSAC 方法

2.2.1 RANSAC 算法简介

RANSAC 为 Random Sample Consensus 的缩写，它是根据一组包含异常数据的样本数据集，计算出数据的数学模型参数，得到有效样本数据的算法。它于 1981 年由 Fischler 和 Bolles 最先提出[4]。RANSAC 算法经常用于计算机视觉中，例如在立体视觉领域中同时解决一对相机的匹配点问题及基本矩阵的计算。

RANSAC 算法具有如下基本假设：

a） 数据由“局内点”组成，即数据的分布可用一些模型参数来描述；

b） “局外点”是不能适应该模型的数据；

c） 除此之外的数据视为噪声。

基于以上假设，可以设计如下算法流程：

a） 从数据中随机选择一组可以拟合模型的数据；

b） 给定阈值误差 ε，检测所有数据是否符合模型，符合即视为内点，否则视为外点；

c） 不断迭代上述步骤，直到获得足够多的内点。

不难发现，RANSAC 算法的核心思想就是在匹配的特征点中随机选取数据，通过计算和不断迭代，寻找到最优的参数模型，在这个最优模型中，能匹配上的特征点最多。

2.2.2 RANSAC 算法特征匹配

RANSAC 算法需要寻找一个最优单应性矩阵 H，矩阵大小为 3×3。利用 RANSAC 算法找到的最优单应性矩阵 H 需要使得满足该矩阵的匹配特征点最多。由于常常令 $h_{33}=1$ 来归一化矩阵，所以单应性矩阵 H 只有 8 个未知参数，所以至少需要 8 个线性方程才能进行求解，而对应到点位置信息上，一组特征匹配点可以得到两个线性方程，因此至少需要 4 组特征匹配点对才能求解得到单应性矩阵 H。

$$s\begin{bmatrix}x'\\y'\\1\end{bmatrix}=\begin{bmatrix}h_{11}&h_{12}&h_{13}\\h_{21}&h_{22}&h_{23}\\h_{31}&h_{32}&h_{33}\end{bmatrix}\begin{bmatrix}x\\y\\1\end{bmatrix} \tag{2-2}$$

式中：(x, y)——目标图像关键点位置，(x', y')——场景图象关键点位置，s——尺度参数。

RANSAC 算法从匹配数据集中随机抽出 4 个样本，并保证这四个样本之间不共线。然后计算出其单应性矩阵，然后利用这个模型测试所有数据，并计算满足这个模型数据点的个数与投影误差（即代价函数）。若此模型为最优模型，则对应的代价函数最小。计算代价函数的公式如下：

$$\sum_{i=1}^{n}\left(x_i' - \frac{h_{11}x_i + h_{12}y_i + h_{13}}{h_{31}x_i + h_{32}y_i + h_{33}}\right)^2 + \left(y_i' - \frac{h_{21}x_i + h_{22}y_i + h_{23}}{h_{31}x_i + h_{32}y_i + h_{33}}\right)^2 \tag{2-3}$$

3. 图像的 SIFT 特征提取与匹配

使用 python 调用 opencv 库实现 SIFT 特征提取与匹配，程序框图如下：

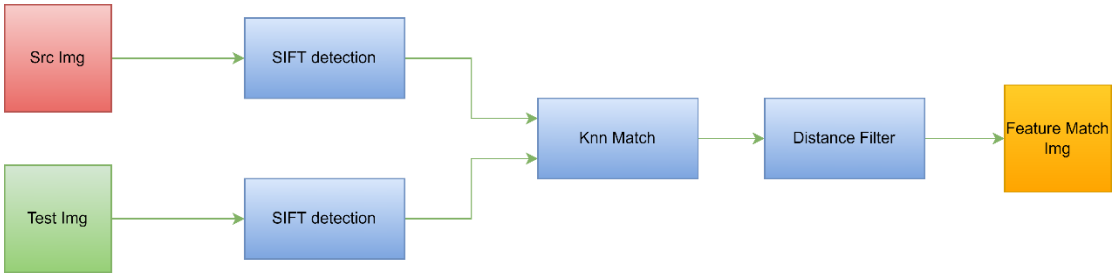


图 3-1，特征提取与匹配程序框图

源图像与测试图像如下：

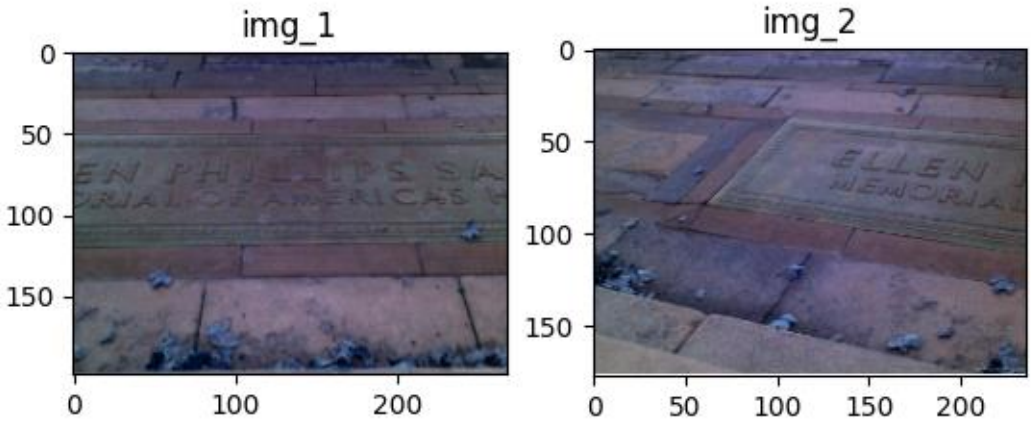


图 3-2， 源图像(img_1)与测试图像(img_2)

SIFT 特征提取图像如下：

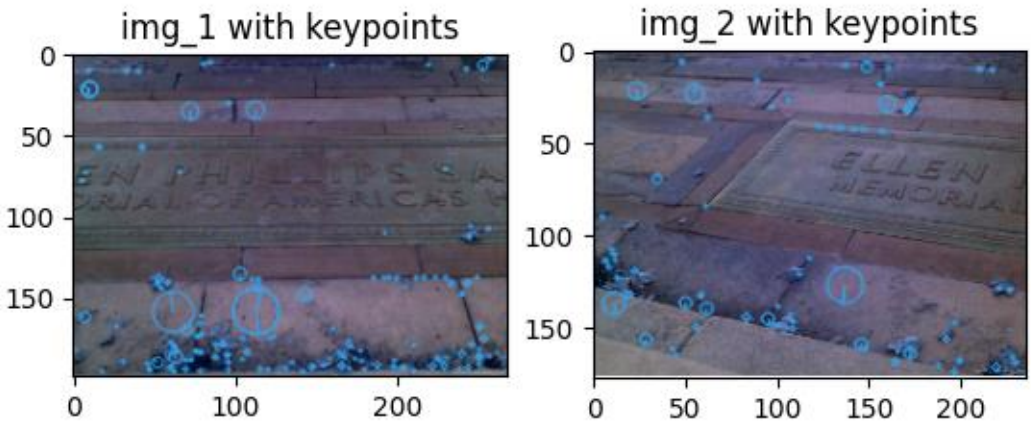


图 3-3，源图像特征图(img_1 with keypoints)与测试图像特征图(img_2 with keypoints)特征粗匹配图像如下：

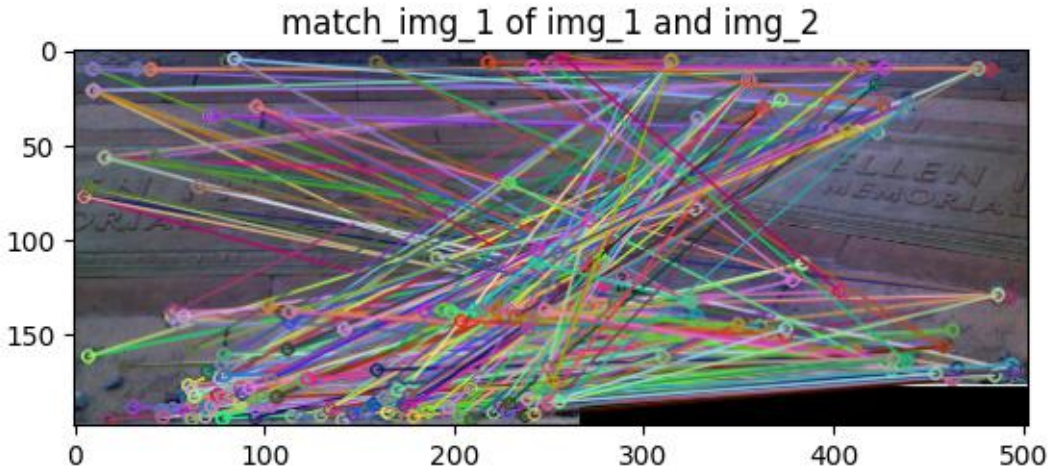


图 3-4，特征粗匹配图像

使用距离度量过滤误匹配点得到匹配图像如下：

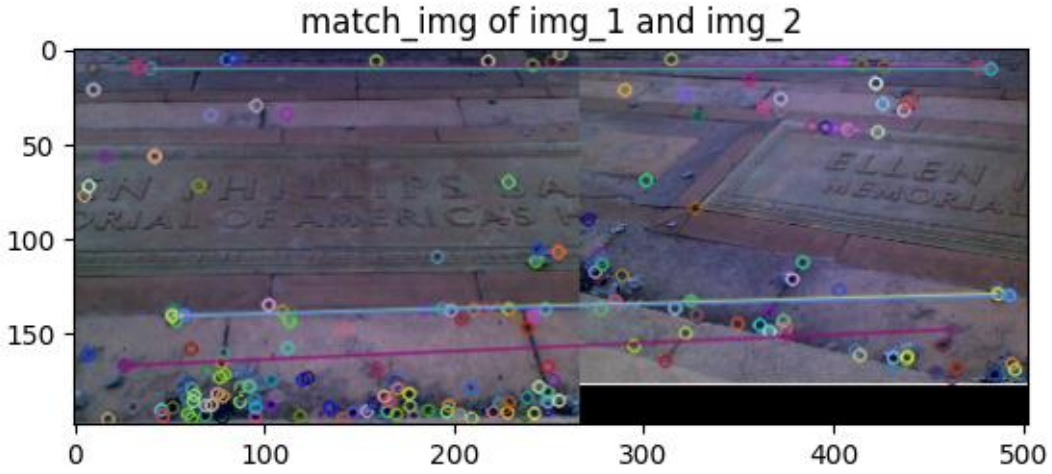


图 3-5，距离度量过滤后的特征匹配图像

这里选择过滤条件如下：

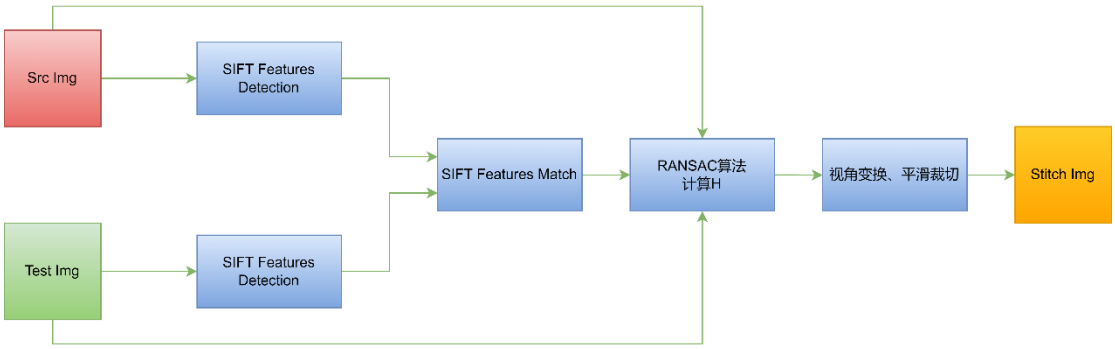
```
if m.distance < 0.5*n.distance:
# 如果第一个邻近距离比第二个邻近距离的 0.5 倍小，则保留
```

得到过滤前和过滤后的匹配点数如下：

match points nums: 150；good match points nums: 5。

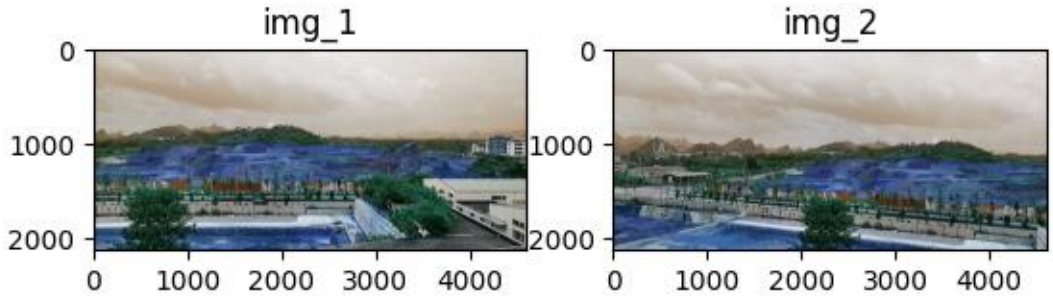
4. 基于单应矩阵 H（2D 射影变换）的图像视点变换与拼接

使用 python 调用 opencv 库实现 SIFT 特征提取与匹配，并使用 RANSAC 算法计算单应矩阵对测试图像进行视角变换，之后进行拼接、平滑以及裁剪操作得到拼接图象。程序框图如下：



4-1，图像拼接程序框图

源图像与测试图像如下：



4-2，源图像(img_1)与测试图像(img_2)

特征提取图像如下：

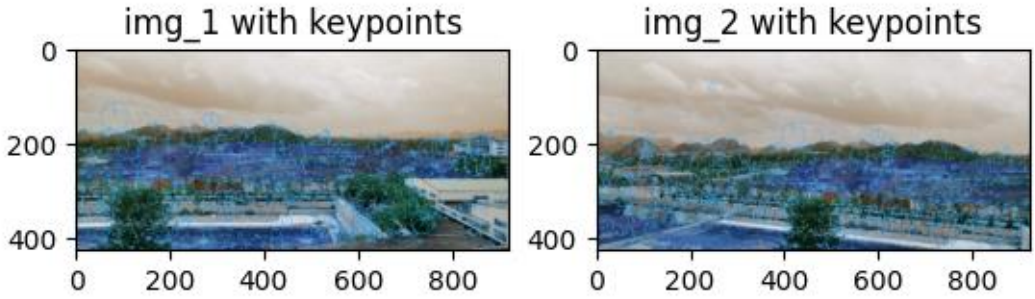


图 4-3，源图像特征图(img_1 with keypoints)与测试图像特征图(img_2 with keypoints)

特征匹配图像如下：

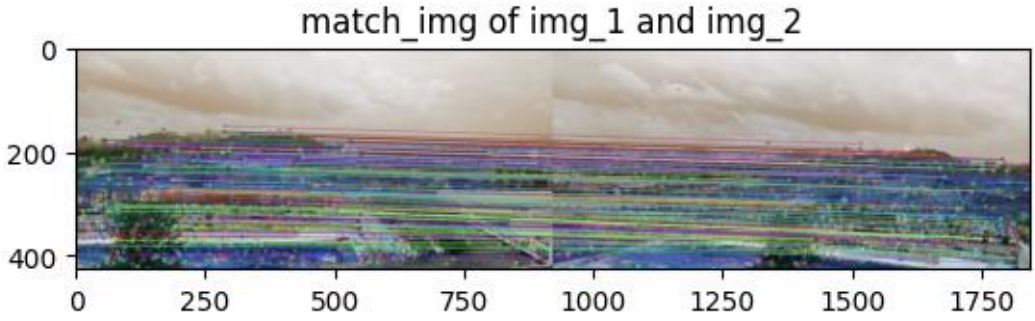


图 4-4，特征匹配图像

计算得到单应矩阵 H 如下：

$$H = \begin{bmatrix} 5.26020366e-01 & -4.06202337e-02 & 3.60513912e+02 \\ -1.00400689e-01 & 8.25595976e-01 & 6.10078046e+01 \\ -4.98975248e-04 & -4.69898201e-05 & 1.00000000e+00 \end{bmatrix} \quad (4\text{-}1)$$

对源图像和测试图像进行视角变换、平滑裁切得到拼接图像如下：

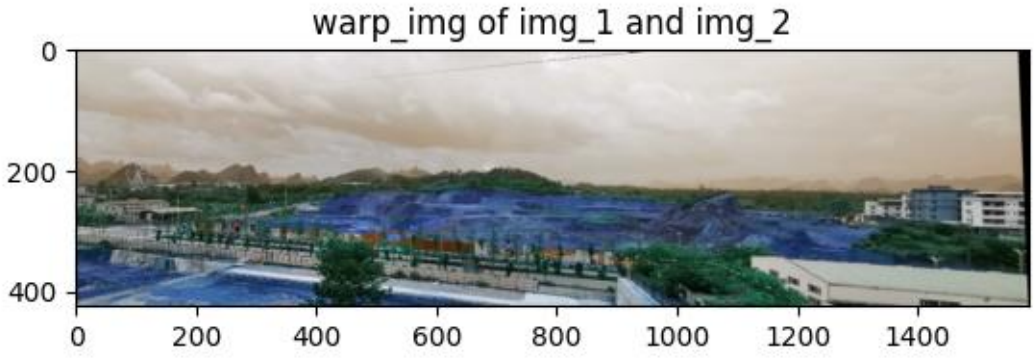


图 4-5，拼接图像

References:

[1] D. G. Lowe, "Object recognition from local scale-invariant features," Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, pp. 1150-1157 vol.2, doi: 10.1109/ICCV.1999.790410.

[2] Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60, 91–110 (2004).

[3] Lindeberg, Tony, "Scale-space theory: A basic tool for analysing structures at different scales", Journal of Applied Statistics, 21, 2 (1994), pp. 224–270.

[4] Robert C. Bolles, Martin A. Fischler， A RANSAC-Based Approach to Model Fitting and Its Application to Finding Cylinders in Range Data. IJCAI 1981: 637-643

Appendix:

Full code also can be found in the below link:

https://github.com/JulyThirteenth/cvpr_assignments/tree/main/Homework-1