

# 基于节点特征对齐的图卷积方法

某某某

## 摘要

这里是摘要

# 目录

1	介绍	II
2	预处理	II
3	种子模块	IV

# 1 介绍

## 2 预处理

### 提取特征点

多种算法比较，在讨论中体现性能。

### 计算匹配点

计算两幅图片的 topN 匹配，选出 N 个初步匹配特征点。计算这几个特征点位置坐标、领域特征的表示和特征点表示共同组成配对点特征。

聚合数量：聚合初匹配特征点的领域半径内选定数量节点，以离该点的距离为评判标准，从近到远选择，不足则复制最近点特征到预定数量，然后再用 self-attention 机制进行聚合。

聚合领域特征：knn，最远点搜索，self-attention,  $\mathbf{X}_n = Att(\mathbf{X}, \mathbf{X}, w)$

1、种子点数量：根据统计量来预定义，统计特征点之间相关性分布。

2、特征点位置向量由一维度卷积网络学习表示生成，暂定 128 维度特征，因为 **SIFT** 等特征算法计算特征点表示多为 128 维。

3、领域半径：领域半径为初匹配后节点的最远距离的 0.1 倍。

坐标系的基本概念：1、世界坐标系：也称之为测量坐标系，是一个三维直角坐标系，以其为基准可以描述相机和待测物体的空间位置。世界坐标系的位置可以根据实际情况自由而定。2、相机坐标系：和相机有关，也是一个三维直角坐标系， $X, Y$  与成像平面平行，原点为光心， $Z$  轴为光轴方向。3、图像坐标系：平面二维坐标系，与成像平面平行，院线在图像中心。4、像素坐标系：以像素为单位，原点在图像左上角。

### 坐标系之间的转换：世界坐标系和相机坐标系的转换

反映相机在世界坐标系中的位置，也叫做外参，用一个旋转的  $R$  和平移  $T$  矩阵表示，转换关系如下：

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

其中  $R$  为 3x3 的旋转矩阵， $t$  为 3x1 的平移向量。

推导过程：

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

为了计算方便，改为齐次坐标表示：

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_1 \\ R_{21} & R_{22} & R_{23} & t_2 \\ R_{31} & R_{32} & R_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

### 相机坐标系 $P_C$ 到像素坐标系 $P_x$ 的转换

像素坐标系相比与相机坐标系进行了缩放和平移，将相机坐标系中的三维坐标转换为像素坐标系中的二维坐标。设相机的焦距为  $f$ ，图像的像素坐标为  $(u, v)$ ，相机坐标系中的点坐标为  $(X, Y, Z)$

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

其中  $f_x$  和  $f_y$  分别表示相机水平和垂直焦距， $c_x$  和  $c_y$  表示相机主点在图像中的像素坐标，水平焦距和垂直焦距分别表示相机的水平和垂直成像范围。

### 图像坐标系和相机坐标系的转换

将图像坐标系转换到相机坐标系需要考虑相机的内参和外参。相机内参包括相机的焦距、像素尺寸等信息，用于描述从相机坐标系到图像坐标系的投影关系。相机外参包括相机在三维空间中的位置和朝向，用于描述从世界坐标系到相机坐标系的变换关系。

以下是从图像坐标系到相机坐标系的转换过程：

首先，需要根据相机内参将图像坐标系中的像素坐标  $(u, v)$  转换为归一化图像平面坐标  $(x, y)$ ，即将像素坐标除以图像的宽和高：

$$x = \frac{u - c_x}{f_x}$$

$$y = \frac{v - c_y}{f_y}$$

其中  $f_x$  和  $f_y$  是相机的焦距， $c_x$  和  $c_y$  是图像中心点的坐标。

然后，将归一化图像平面坐标  $(x, y)$  和深度  $d$  转换为相机坐标系下的坐标  $(X_c, Y_c, Z_c)$ ：

$$X_c = x \cdot d$$

$$Y_c = y \cdot d$$

$$Z_c = d$$

最后，将相机坐标系下的坐标  $(X_c, Y_c, Z_c)$  转换为世界坐标系下的坐标  $(X_w, Y_w, Z_w)$ ，需要根据相机的外参进行变换：

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}$$

其中， $R$  和  $T$  是相机的旋转矩阵和平移向量，表示从世界坐标系到相机坐标系的变换关系。

需要注意的是，在进行这个转换过程时，需要确保图像坐标系和相机坐标系以及世界坐标系使用的是同样的坐标系。

### 像素坐标系到图像坐标系

在计算机图形学中，像素坐标系是图像坐标系的一种特殊形式，它将图像划分成一个个的像素单元，并使用像素的行列坐标来表示像素在图像中的位置。

将像素坐标系转换为图像坐标系，通常需要考虑以下两个因素：

1、像素的坐标原点位置在像素坐标系中，通常将左上角的像素的坐标设为  $(0, 0)$ ，而在图像坐标系中，通常将左下角的像素的坐标设为  $(0, 0)$ 。因此，需要根据图像的高度将像素坐标系中的坐标转换为图像坐标系中的坐标。

设图像的高度为  $h$ ，像素坐标为  $(u, v)$ ，则可以将其转换为图像坐标系中的坐标  $(x, y)$ ，公式如下：

$$x = u$$

$$y = h - v - 1$$

2、像素的坐标单位在图像坐标系中，通常使用实际长度单位（例如毫米）来表示图像中的像素位置，而在像素坐标系中，通常使用像素单位来表示像素位置。因此，需要根据图像的像素密度将像素坐标系中的坐标转换为图像坐标系中的坐标。

设图像的宽度为  $w$ ，像素坐标为  $(u, v)$ ，每个像素的宽度为  $s$ ，则可以将其转换为图像坐标系中的坐标  $(x, y)$ ，公式如下：

$$x = u \cdot s$$

$$y = (h - v - 1) \cdot s$$

其中  $h$  是图像的高度， $s$  是每个像素的宽度。

需要注意的是，在进行这个转换过程时，需要考虑到像素坐标系和图像坐标系的原点位置和单位长度的不同。

Sinkhorn 算法是一种用于求解最优传输问题（Optimal Transport）的迭代算法，也称为 Sinkhorn-Knopp 算法、Bregman 投影算法或者自对偶重正化（Self-Dual Proximal Gradient）算法。

最优传输问题是指，给定两个概率分布  $\mu$  和  $\nu$ ，找到一个满足一定条件的最优映射  $\gamma$ ，使得映射后的分布与目标分布  $\nu$  之间的距离最小。最优传输问题的求解在图像处理、机器学习、计算几何等领域都有广泛的应用。

Sinkhorn 算法的基本思路是通过迭代，不断逼近最优映射，直至收敛为止。算法的具体步骤如下：

初始化：给定两个概率分布  $\mu$  和  $\nu$ ，初始化一个矩阵  $K$ ，其中  $K_{i,j} = \exp(-c_{i,j})$ ， $c_{i,j}$  是  $\mu$  中第  $i$  个元素和  $\nu$  中第  $j$  个元素之间的距离，可以是欧几里得距离、马氏距离等。

迭代更新：重复执行以下步骤，直至收敛：

- a. 计算  $\gamma$ ： $\gamma_{i,j} = \frac{K_{i,j} u_i v_j}{\epsilon + \sum_{i',j'} K_{i',j'} u_{i'} v_{j'}}$ ，其中  $u$  和  $v$  是两个向量，分别表示从  $\mu$  和  $\nu$  出发的概率质量。
- b. 更新  $u$  和  $v$ ： $u_i = \frac{\mu_i}{\sum_j \gamma_{i,j}}$ ， $v_j = \frac{\nu_j}{\sum_i \gamma_{i,j}}$ 。

输出  $\gamma$ 。

其中， $\epsilon$  是一个较小的正数，用于避免分母为零。

可以将 Sinkhorn 算法理解为一种自对偶的优化算法，每次迭代都会对偶问题和原始问题进行更新，同时还使用了 Bregman 投影和次梯度技巧加速收敛。该算法的时间复杂度为  $O(n^2 \log n)$ ，具有较好的可扩展性和数值稳定性。

### 3 种子模块

得到两幅图片特征点后计算单应矩阵，将原图片四个角转换到目标图片对应坐标计算两幅图片的重叠部分。排除掉不在重叠区域的特征点，当然可以考虑增加 over 区域。