

文章编号:1007-5321(2021)06-0013-07

DOI:10.13190/j.jbupt.2021-072

一种面向弱纹理图像的特征点描述子

程鹏飞¹, 周修庄², 唐玲², 魏世民¹, 高欢¹

(1. 北京邮电大学 现代邮政学院, 北京 100876; 2. 北京邮电大学 人工智能学院, 北京 100876)

摘要: 现有的大多数特征点提取算法适用于处理纹理丰富的图像, 而对于弱纹理图像则无法提取有效的特征点。对此, 提出了多邻域结构张量特征(MNSTF)算法。基于一系列固定的邻域和图像结构张量, 通过表达局部图像的结构和纹理信息, 解决了弱纹理和无纹理场景下特征点提取和匹配等相关问题; 同时, 通过计算邻域之间的相对方向, 实现了 MNSTF 算法特征描述子的旋转不变性。实验结果表明, MNSTF 算法在经过旋转的弱纹理图像测试集上的特征点匹配准确率达到 99.9% 以上, 验证了其良好的适用性、旋转不变性和鲁棒性。

关键词: 特征点描述子; 多邻域; 结构张量; 弱纹理图像

中图分类号: TP391.4

文献标志码: A

A Feature Point Descriptor for Texture-Less Images

CHENG Peng-fei¹, ZHOU Xiu-zhuang², TANG Ling², WEI Shi-min¹, GAO Huan¹

(1. School of Modern Post, Beijing University of Posts and Telecommunications, Beijing 100876, China;

2. School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: Most of the existing feature point extraction algorithms are suitable for processing images with rich texture, but they cannot extract effective feature points for texture-less images. To solve this problem, a new feature point descriptor, termed as multi-neighborhood structure tensor features (MNSTF) is proposed. The algorithm is based on a series of fixed neighborhoods and image structure tensors. By expressing the structure and texture information of local images, it solves the problems such as feature point extraction and matching in texture-less and non-texture scenes in existing feature point extraction algorithms. At the same time, by calculating the relative direction between neighborhoods, the rotation invariance of the MNSTF feature descriptor. The experimental results show that the accuracy of feature point matching of the MNSTF algorithm on the rotated texture-less image test set is over 99.9%, which verifies the good applicability, rotation invariance and robustness of the MNSTF feature descriptor.

Key words: feature point descriptor; multi-neighborhood; structure tensor; texture-less images

特征点提取算法的主要目的是将原始图像投影变换到一个低维特征空间, 得到最能够反映特征点本质或区分不同特征点的低维向量, 它广泛应用于物体定位与识别、三维重建、图像检索、相

机标定等领域。常见的物理对象通常包含弱纹理区域, 颜色单一, 纹理重复, 点特征少, 像素之间的区分性较差。弱纹理图像的特征点提取是计算机视觉领域的一个技术难点, 现有的一些特征点算

收稿日期: 2021-04-28

基金项目: 国家自然科学基金项目(61972046); 北京市自然科学基金项目(4202051)

作者简介: 程鹏飞(1995—), 男, 硕士生。

通信作者: 唐玲(1986—), 女, 高级工程师, 硕士生导师, E-mail: tangling@bupt.edu.cn.

法^[1-5]在实际应用中取得了良好效果,然而对于一些弱纹理以及无纹理的图像,这些算法提取到的特征点数量较少,应用效果不够理想.为了解决弱纹理、无纹理场景下特征提取的问题,很多研究者提出了基于模板的方法. Tombari 等^[6]提出了 BOLD(bunch of lines descriptor)算法,使用线段特征解决了弱纹理物体识别的问题; Chan 等^[7]介绍了 BORDER(bounded oriented rectangle descriptors for enclosed region)算法,使用矩形封装对象,检测最小的离群值; Hinterstoisser 等^[8-9]采用多模态的方法解决了弱纹理下的目标位姿估计问题; Line-HOG 算法^[10]基于线段的特征描述子,通过对线段进行编码,解决了丰富纹理和弱纹理图像的目标检测问题.然而,这些方法只能解决目标检测这类特定的问题,无法广泛使用于视觉领域.

随着深度学习技术的发展,利用神经网络解决弱纹理问题的方法相继被提出. Chan 等^[11]提出了 BIND(binary integrated net descriptor)算法,这是一种具有多层结构的二进制网络,可检测弱纹理物体的检测器;为了提高物体检测的速度, Yu 等^[12]引入了方向压缩图和二进制字符串描述,具有良好的检测效果; Fang 等^[13]提出了一种称为 DOG 的方法,在卷积神经网络的基础上,有效地去除了输入图像的背景,提高了分类器的准确率; Kaskman 等^[14]将弱纹理物体的空间计算机辅助设计模型和彩色图像用于神经网络的训练,获取目标物体的 3D 点的特征,实现了弱纹理物体的检测与位姿估计; Jiang 等^[15]通过卷积神经网络完成对弱纹理物体的抓取.然而,用神经网络的方法解决弱纹理问题需要大量精确标注的训练样本和昂贵的机器,难以被广泛使用.

笔者提出了多邻域结构张量特征(MNSTF, multi-neighborhood structure tensor features)算法.主流的特征点算法大都只依赖图像的纹理信息,却忽略了图像的结构信息,因而无法解决无纹理的问题. MNSTF 算法则通过一系列固定的特征点邻域以及图像局部结构张量,使得 MNSTF 算法的特征描述子不仅表达了纹理信息,也表达了图像结构层次的信息,实现了 MNSTF 算法的旋转不变性、高可判别性以及鲁棒性,解决了弱纹理和无纹理图像的特征点提取与匹配的问题.

1 多邻域方法

在现实世界中,图像特征点的识别大都可以通

过人眼直接或间接完成,而现有的各种特征点算法却还没有达到这种能力.究其原因,是大多特征点算法没有获取足够的信息,仅仅根据一个特征点邻域内的纹理信息提取特征描述子,如果特征点邻域内的信息不足以区分特征点时,算法就失去了效果.为了解决这个问题,笔者提出了多邻域的方法,该方法对于无纹理、弱纹理的场景仍然可以提取足够多的有效特征点,为相关的应用场景提供解决方案.

下面以图 1 为例说明多邻域的方法. 图 1 中标注的 A 点和 B 点是目标点,以 A 点和 B 点为中心分别有 2 个不同大小的圆形邻域 I_1 和 I_2 ,其中邻域 I_1 的直径小于邻域 I_2 的直径. 当 A 点和 B 点的邻域为 I_1 时, A 点和 B 点邻域内的像素信息是相同的, A 点和 B 点所在的线段都可以通过旋转与另一条重合,此时无论是算法还是人眼都无法对这 2 点进行区分;当 A 点和 B 点的邻域扩大到 I_2 时, A 点和 B 点邻域内的像素分布已经有了很大不同,邻域内的像素信息不再相同,可以尝试通过算法对这 2 个特征点进行区分,但仅扩大特征点的邻域范围,并不能表达图像的结构信息,而且可能会引入误差信息.因此,这里采用多个邻域的方法,通过对不同邻域之间的关系进行建模,从而获得特征点的特征描述子,实现图像结构信息与纹理信息的表达.

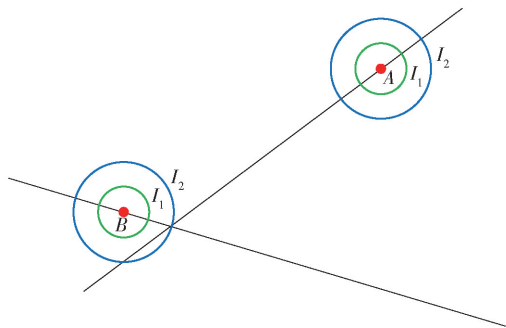


图1 多邻域方法

2 MNSTF 算法

基于多邻域方法提出了 MNSTF 算法,通过计算一系列邻域 $I = [I_1, I_2, \dots, I_i, \dots, I_m]$ 结构张量的特征值和特征向量,形成了该算法的特征描述子,实现旋转不变性、高可判别性、通用性以及鲁棒性.以结构张量为工具,对特征点邻域像素进行数学形式的表达,并利用相对思想实现了 MNSTF 算法的旋转不变性,最后得到表达图像纹理和结构信

息的特征点描述子. 以下将分别介绍 MNSTF 算法的局部结构张量、实现旋转不变性以及获得特征描述子的方法.

2.1 局部结构张量

结构张量经常被用于各向异性过滤^[16-17]、运动检测^[18]等,基于图像的梯度信息,可以确定图像的边缘点和角点^[19],而 MNSTF 算法使用结构张量方法的目的是度量特征点邻域内梯度的分布. 定义弱纹理图像的局部结构张量为

$$T_{\sigma} = \begin{bmatrix} g_x^2 G_{\sigma} & g_x g_y G_{\sigma} \\ g_y g_x G_{\sigma} & g_y^2 G_{\sigma} \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} \quad (1)$$

其中: G_{σ} 是标准差为 σ 的高斯函数, g_x 和 g_y 分别为每个像素的水平梯度和垂直梯度. 因为 T_{σ} 是对称和半正定的,所以它有 2 个正交的特征向量:

$$e_1 = \begin{pmatrix} T_{22} - T_{11} + \sqrt{(T_{22} - T_{11})^2 + 4T_{12}^2} \\ -2T_{12} \end{pmatrix} \quad (2)$$

$$e_2 = \begin{pmatrix} 2T_{12} \\ T_{22} - T_{11} + \sqrt{(T_{22} - T_{11})^2 + 4T_{12}^2} \end{pmatrix} \quad (3)$$

相应的特征值为

$$\lambda_1 = \frac{1}{2}(T_{22} + T_{11} - \sqrt{(T_{22} - T_{11})^2 + 4T_{12}^2}) \quad (4)$$

$$\lambda_2 = \frac{1}{2}(T_{22} + T_{11} + \sqrt{(T_{22} - T_{11})^2 + 4T_{12}^2}) \quad (5)$$

特征值越大,表示图像梯度分布在对应特征向量方向占的比重越大. 这里 $\lambda_2 > \lambda_1$,所以 λ_2 为主特征值,则 e_2 为主特征向量. 基于特征点邻域结构张量的特征向量,进一步实现特征描述子的旋转不变性.

2.2 旋转不变性

旋转不变性是衡量特征算法性能的一个重要指标. MNSTF 算法通过计算邻域之间相对方向的方法,使特征点描述子具有良好的旋转不变性.

MNSTF 算法实现旋转不变性的方法和步骤如下:首先,选择一个合适大小的邻域作为基邻域 I_b ,计算 I_b 结构张量的主特征向量,并作为基特征向量 e_b ;然后,计算其他邻域的特征向量的角度与 e_b 向量角度的差值,以实现 MNSTF 算法的旋转不变性. 假设式(2)和式(3)计算得到的特征向量为

$$e_i = \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} \quad (6)$$

则由特征向量 e_i 所确定的角度为

$$\theta_i = \begin{cases} \arctan\left(\frac{\beta_2}{\beta_1}\right), & \beta_1 > 0 \\ \arctan\left(\frac{\beta_2}{\beta_1}\right) + \pi, & \beta_1 < 0 \\ \frac{\pi}{2}, & \beta_1 = 0 \text{ and } \beta_2 > 0 \\ -\frac{\pi}{2}, & \beta_1 = 0 \text{ and } \beta_2 < 0 \end{cases} \quad (7)$$

选择合适大小的特征点邻域,确定特征点的基础角度 θ_b ,则角度差值为

$$\Delta\theta = \theta_b - \theta_i \quad (8)$$

由于邻域与邻域之间计算得到的角度差值并不会随着图像的旋转发生变化,进而可以根据该角度差值实现特征描述子的旋转不变性. 在计算得到邻域角度差值后,将其转化为向量形式,并进行标准化的处理,计算如下:

$$D_i = \left[\frac{1}{\sqrt{1 + \tan^2(\Delta\theta)}}, \frac{\tan(\Delta\theta)}{\sqrt{1 + \tan^2(\Delta\theta)}} \right] \quad (9)$$

这里将向量 D_i 称为差值特征向量,旋转后的同一个特征点的差值特征向量近似相等.

2.3 MNSTF 算法的特征描述子

特征描述子代表了兴趣点邻域的特征信息,用来区分不同的点. MNSTF 算法通过计算一组特征点邻域 $I = [I_1, I_2, \dots, I_i, \dots, I_m]$ 结构张量的特征值和特征向量,经过变换后组成一个长度与邻域个数相关的向量. 通过对结构张量的特征向量进行处理,得到了差值特征向量,实现了 MNSTF 算法的旋转不变性,为了保证特征描述子的高可判别性,还需要在差值特征向量中融合特征值.

图像旋转之后,特征点邻域结构张量的特征值大小会发生变化,笔者通过计算比例特征值 R_i 解决这个问题. 选择合适大小的邻域 I_b 确定特征点邻域的结构张量的基特征值 λ_b ,其余邻域得到的结构张量特征值 λ_i 通过与 λ_b 相除得到比例特征值:

$$R_i = \frac{\lambda_i}{\lambda_b} \quad (10)$$

将通过式(9)和式(10)计算得到的差值特征向量和比例特征值相乘,得到局部 MNSTF 算法特征描述子:

$$f_i = R_i D_i = \left[\frac{\lambda_i}{\lambda_b \sqrt{1 + \tan^2(\Delta\theta)}}, \frac{\lambda_i \tan(\Delta\theta)}{\lambda_b \sqrt{1 + \tan^2(\Delta\theta)}} \right] \quad (11)$$

将不同邻域确定的局部 MNSTF 算法特征组合

成一个向量,即 MNSTF 算法描述子,有

$$\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_i, \dots, \mathbf{f}_m] \quad (12)$$

值得注意的是,1 个特征点邻域的结构张量可以得到 2 对特征值和特征向量,MNSTF 算法根据特征值的大小,对这 2 对特征值和特征向量进行排序,以确定 MNSTF 算法特征描述子内的局部 MNSTF 算法特征顺序。

3 实验与分析

选择 6 张无纹理、弱纹理以及丰富纹理的典型图像作为测试集进行实验,其中图 2(a)为无纹理图像,图 2(b)和(f)为弱纹理图像,图 2(c)~(e)为丰富纹理图像,如图 2 所示。

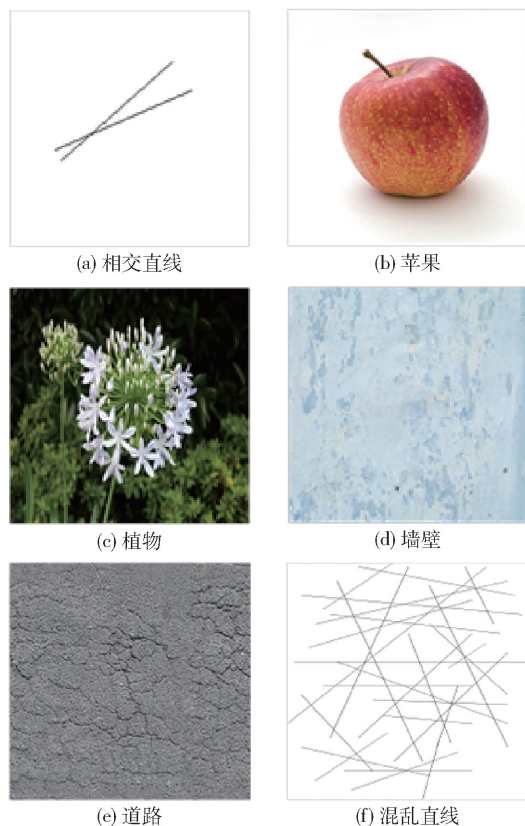


图 2 测试图像

3.1 MNSTF 特征匹配实验

对图 2 中的 6 张图片进行了 MNSTF 算法的特征匹配实验,以验证 MNSTF 算法的高可判别性. 首先将图片旋转一定的角度,得到新的图片;然后对原图片和旋转后的图片应用 MNSTF 算法提取特征并进行匹配. 由于匹配成功的特征点较多,为了能够直观地看出算法的实际效果,实验结果中特征点数量只以 1:50 的比例进行显示,如图 3 所示。

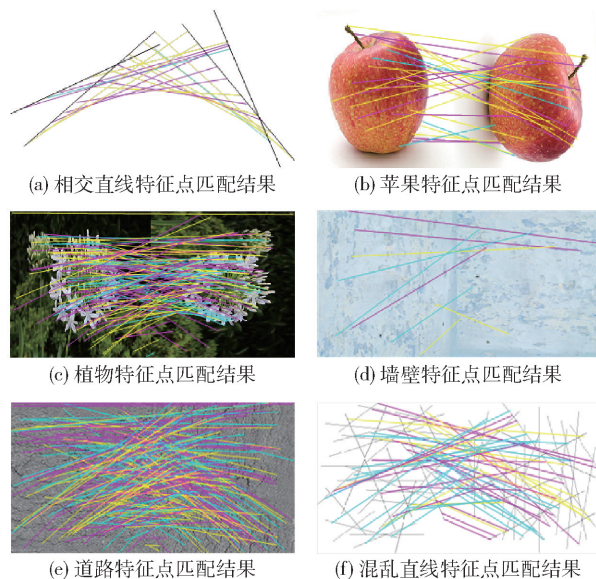


图 3 MNSTF 算法特征点匹配测试结果

由图 3 可以看出,无论是丰富纹理的图像,还是弱纹理、无纹理的图像,使用 MNSTF 算法都能够提取到足够多的特征点,且具有较高的可判别性,可达到一定的特征点匹配准确率。

3.2 MNSTF 算法与 SIFT 算法的对比实验

为了进一步了解 MNSTF 算法的性能,以图 2 中的测试图像为基础,对 MNSTF 和 SIFT (scale invariant feature transform)^[1] 算法进行了定量分析. SIFT 算法是特征点算法中应用最为广泛的,在可判别性、鲁棒性、适用性等方面代表了特征点算法最高的水平^[20]. 目前有较多的 SIFT 改进算法^[21],但是并不适用于弱纹理图像,因此实验中将 MNSTF 算法与经典 SIFT 算法进行了比较。

考虑到 SIFT 算法应用图像金字塔实现不同尺度图像的特征点检测,最后的特征点定位存在误差,因此实验结果分别以 0、1、5 像素的匹配允许误差给出。

SIFT 算法过于依赖局部的纹理信息,当图像中没有明显的纹理时,SIFT 算法就无法提取有效的特征点. 从表 1~表 3 可以看出,SIFT 算法无法处理类似图 1(a)所示的弱纹理图像,因为其太过于依赖局部的纹理信息. 当图像中没有明显的纹理时,SIFT 算法就无法提取有效的特征点. 多邻域的方法使 MNSTF 算法描述子不再完全依赖一个固定邻域的纹理信息,而能够同时表达图像纹理与结构信息. 从 MNSTF 算法的测试结果可以发现,MNSTF 算法对无纹理物体仍然可以提取足够多的特征点,且特

征点的匹配准确率可以达到 77.42%。

从特征点的数量来看,对于各种类型的图像,MNSTF 算法提取到的特征点数量都远远大于 SIFT 算法。SIFT 算法把物体边缘点当作不稳定的点,所以在兴趣点检测时会把边缘点剔除,这也使 SIFT 算法能够提取到的特征点非常少。因为 MNSTF 算法加入了图像的结构信息,可以认为边缘点具有足够高的判别性。实验结果也证实了这一点。

相较于 SIFT 算法,MNSTF 算法的特征点定位会更加精准。当特征点匹配容许误差为 0 时,SIFT 算法只有50%左右的匹配准确率,只有扩大特征点

匹配容许误差,才能表现出 SIFT 算法良好的效果,这是因为其中使用了图像金字塔,限制了 SIFT 算法在一些对精度要求较高场景的应用。当 MNSTF 算法特征点匹配的允许误差为 0 个像素时,图 2(a)所示无纹理图像的特征点匹配准确率为 77.42%,而其他弱纹理或丰富纹理图像的特征点匹配准确率可以达到 99.9%,具有较高的特征点定位精度及匹配准确率。

3.3 MNSTF 算法与其他特征点算法的对比实验

目前经典的 SIFT 算法仍然广泛应用于工业领域,但近年也有效果良好的特征点算法相继被提出。基于 SIFT 算法的 SURF (speeded up robust features)^[2]算法、BRIEF (binary robust independent elementary features)^[3]算法、基于 FAST (the features from accelerated segment test)^[5]算法和 BRIEF 算法的 ORB (oriented fast and rotated brief)^[4]特征点算法是目前主流的特征点算法。将 MNSTF 算法与 SURF 算法、BRIEF 算法和 ORB 算法进行实验对比,验证了所提算法的有效性。

SURF 算法实现了特征金字塔的方法,定位较准确,而 BRIEF 和 ORB 算法没有实现尺度不变性,因此该部分特征点匹配实验只在位置匹配容许误差为 0 时进行,实验结果如表 4 和表 5 所示。

表 1 匹配容许误差为 0 时 MNSTF 与 SIFT 算法的实验结果

图像	MNSTF 算法特征点个数	SIFT 算法特征点个数	MNSTF 算法特征点匹配准确率/%	SIFT 算法特征点匹配准确率/%
相交直线	1 116	—	77.42	—
苹果	1 762	63	99.94	47.62
花朵	5 412	602	100.00	52.66
墙壁	528	52	100.00	65.38
道路	10 736	47	99.98	46.81
混乱直线	2 836	40	100.00	55.00

注：—表示该算法对相应的图像无效

表 2 匹配容许误差为 1 时 MNSTF 与 SIFT 算法的实验结果

图像	MNSTF 算法特征点个数	SIFT 算法特征点个数	MNSTF 算法特征点匹配准确率/%	SIFT 算法特征点匹配准确率/%
相交直线	1 116	—	77.42	—
苹果	1 762	63	99.94	87.30
花朵	5 412	602	100.00	97.34
墙壁	528	52	100.00	100.00
道路	10 736	47	99.98	100.00
混乱直线	2 836	40	100.00	100.00

注：—表示该算法对相应的图像无效

表 3 匹配容许误差为 5 时 MNSTF 与 SIFT 算法的实验结果

图像	MNSTF 算法特征点个数	SIFT 算法特征点个数	MNSTF 算法特征点匹配准确率/%	SIFT 算法特征点匹配准确率/%
相交直线	1 116	—	77.42	—
苹果	1 762	63	99.94	90.48
花朵	5 412	602	100.00	99.83
墙壁	528	52	100.00	100.00
道路	10 736	47	99.98	100.00
混乱直线	2 836	40	100.00	100.00

注：—表示该算法对相应的图像无效

表 4 不同算法提取的特征点数量

图像	MNSTF	SURF	BRIEF	ORB
相交直线	1 116	5	7	174
苹果	1 762	28	22	311
花朵	5 412	253	242	407
墙壁	528	1	113	179
道路	10 736	6	424	328
混乱直线	2 836	22	119	321

表 5 不同算法的特征点匹配准确率 %

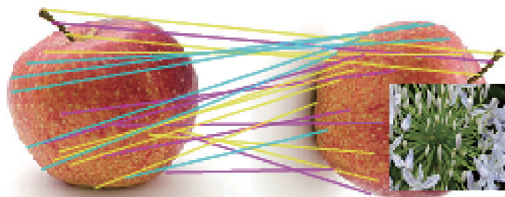
图像	MNSTF	SURF	BRIEF	ORB
相交直线	77.42	100.00	0.00	97.13
苹果	99.94	100.00	0.00	100.00
花朵	100.00	94.47	0.41	99.51
墙壁	100.00	100.00	0.88	100.00
道路	99.98	100.00	0.94	100.00
混乱直线	100.00	100.00	0.00	100.00

由表 4 可知,MNSTF 算法提取到的特征点数量要远远大于其他的特征点算法,这一特性使 MNSTF 算法更加适合于某些应用场景。同时,从表 5 可以看出,在相交直线图像测试中,MNSTF 算法的特征点匹配准确率低于 SURF 和 ORB 算法;在其他图像

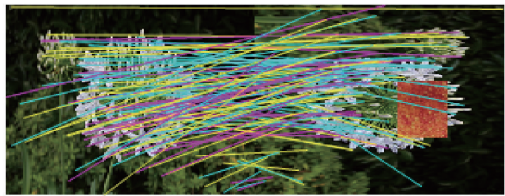
的测试中,MNSTF 算法的特征点匹配的准确率均大于或等于其他算法. 其中由于 BRIEF 算法不具有旋转不变性,特征点的匹配准确率较低. 总体来说,MNSTF 算法在特征点匹配数量上具有较大优势,且具有较好的特征点匹配准确率.

3.4 MNSTF 算法遮挡实验

在很多工作环境中,遮挡问题不可避免,处理遮挡的能力也是用来衡量特征算法优劣的一个指标. 为了验证 MNSTF 算法处理遮挡问题的能力,实验中选取图 2(b)和(c)所示的图像进行遮挡处理,实验结果如图 4 所示.



(a) 苹果遮挡实验结果



(b) 植物遮挡实验结果

图 4 MNSTF 算法遮挡实验

由实验结果可知,未遮挡区域能够很好地与原图进行匹配,验证了 MNSTF 算法解决遮挡问题的有效性.

3.5 参数敏感性实验分析

MNSTF 算法是基于一系列变化的邻域实现的,因此特征点邻域的选择对 MNSTF 算法的性能具有较大的影响. 邻域过小,偶然性会很大;邻域过大,则会加入很多干扰信息. 选择图 2(b)所示的苹果和图 2(c)所示的植物作为测试图像进行实验,分析起始邻域半径以及特征点邻域间隔对特征点数量、特征点匹配准确率的影响. 实验结果如表 6 和表 7 所示.

从表 6 和表 7 可以看出,选择不同的起始邻域半径以及邻域之间的间隔对 MNSTF 算法的性能具有一定的影响,选择合适的参数可以提高 MNSTF 算法特征点的数量和特征点的匹配准确率. 对于不同类型的图像,参数选择的策略是:对于纹理比较丰富的图像,可以减少邻域个数,增大邻域之间的像素

间隔,从而提高算法运行的速度;对于纹理较少的图像,可以增加邻域的个数,减少邻域之间的像素间隔,从而提升算法特征点匹配的准确性.

表 6 图 2(b)所示的算法参数敏感性实验结果

起始邻域 半径/像素	邻域间隔 距离/像素	邻域个数	特征点 数量	特征点匹配 准确率/%
1	1	10	168	71.43
2	2	10	901	54.50
1	2	10	1 430	95.45
3	2	10	1 749	99.31
2	3	10	1 762	99.94
1	2	5	1 344	100.00
3	2	5	1 727	100.00
2	3	5	1 762	99.94
2	3	3	1 762	99.94
1	2	3	1 345	100.00
3	2	3	1 728	99.94

表 7 图 2(c)所示的算法参数敏感性实验结果

起始邻域 半径/像素	邻域间隔 距离/像素	邻域个数	特征点 数量	特征点匹配 准确率/%
1	1	10	6	100.00
2	2	10	3 390	100.00
2	3	10	5 412	100.00
1	2	5	5 414	100.00
3	2	5	6 596	99.91
2	3	5	6 773	99.70
1	2	3	5 417	99.96
3	2	3	6 597	99.89
2	3	3	6 674	99.70

4 结束语

介绍了多邻域结构张量特征描述子,解决了弱纹理图像特征点的提取和匹配问题. 实验结果表明,MNSTF 算法具有以下特点:1)良好的适用性. 不仅适用于丰富纹理的图像,也能够处理弱纹理、无纹理的图像,且相对于其他特征点算法,能够提取更多的特征点;2)良好的旋转不变性. 对于旋转后的图像,MNSTF 算法仍然可以将相应的点进行正确的匹配;3)高判别性. 除了相交直线图,测试集中其余图像的特征点匹配准确率在 99.9% 以上;4)较强的鲁棒性. 能够很好地处理遮挡问题.

当然,MNSTF 算法也存在一些缺点,主要包括:1)特征点算法的尺度不变性可以通过图像金字塔的方法实现,但对 MNSTF 算法的实时性有较大的影响,寻求一种合适的尺度不变性策略是研究的重点;2)由于 MNSTF 算法提取到的特征点较多,匹配消

耗的时间较长,相较现有的很多特征点算法,总体的实时性较差。只考虑单个特征点所消耗平均时间的情况,MNSTF算法要优于大部分的特征点算法,提高MNSTF算法的实时性也是未来研究目标之一。

参考文献:

- [1] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110.
- [2] Bay H, Tuytelaars T, van Gool L. SURF: speeded up robust features[M]//Computer Vision-ECCV 2006. Berlin: Springer Berlin Heidelberg, 2006: 404-417.
- [3] Calonder M, Lepetit V, Strecha C, et al. BRIEF: binary robust independent elementary features[M]//Computer Vision-ECCV 2010. Berlin: Springer Berlin Heidelberg, 2010: 778-792.
- [4] Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF[C]//2011 International Conference on Computer Vision. Barcelona: IEEE Press, 2011: 2564-2571.
- [5] Rosten E, Drummond T. Machine learning for high-speed corner detection[M]//Computer Vision-ECCV 2006. Berlin: Springer Berlin Heidelberg, 2006: 430-443.
- [6] Tombari F, Franchi A, Di L. BOLD features to detect texture-less objects[C]//2013 IEEE International Conference on Computer Vision. Sydney: IEEE Press, 2013: 1265-1272.
- [7] Chan J, Lee J A, Qian Kemao. BORDER: an oriented rectangles approach to texture-less object recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE Press, 2016: 2855-2863.
- [8] Hinterstoisser S, Holzer S, Cagniat C, et al. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes[C]//2011 International Conference on Computer Vision. Barcelona: IEEE Press, 2011: 858-865.
- [9] Hinterstoisser S, Cagniat C, Ilic S, et al. Gradient response maps for real-time detection of textureless objects[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(5): 876-888.
- [10] Cvetković S, Grujić N, Ilić S, et al. Detection of texture-less objects by line-based approach[J]. Facta Universitatis, Series: Automatic Control and Robotics, 2020, 18(2): 79-94.
- [11] Chan J, Lee J A, Qian Kemao. BIND: binary integrated net descriptors for texture-less object recognition[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu: IEEE Press, 2017: 3020-3028.
- [12] Yu Hancheng, Qin Haibao, Peng Maoting. A fast approach to texture-less object detection based on orientation compressing map and discriminative regional weight[J]. Algorithms, 2018, 11(12): 201-217.
- [13] Fang Wei, Ding Yewen, Zhang Feihong, et al. DOG: a new background removal for object recognition from images[J]. Neurocomputing, 2019, 361: 85-91.
- [14] Kaskman R, Shugurov I, Zakharov S, et al. 6 DoF pose estimation of textureless objects from multiple RGB frames[C]//Computer Vision-ECCV 2020 Workshops. [S.l.]: Springer, 2020: 612-630.
- [15] Jiang Ping, Ishihara Y, Sugiyama N, et al. Depth image-based deep learning of grasp planning for texture-less planar-faced objects in vision-guided robotic Bin-picking[J]. Sensors, 2020, 20(3): 706-729.
- [16] Weickert J. Anisotropic diffusion in image processing[M]. Stuttgart: Teubner, 1998: 56-57.
- [17] Kuhne G, Weickert J, Schuster O, et al. A tensor-driven active contour model for moving object segmentation[C]//Proceedings 2001 International Conference on Image Processing. Thessaloniki: IEEE Press, 2001: 73-76.
- [18] Wang Zhou, Bovik A C. Modern image quality assessment[J]. Synthesis Lectures on Image, Video, and Multimedia Processing, 2006, 2(1): 1-156.
- [19] Zhang Weichuan, Sun Changming. Corner detection using multi-directional structure tensor with multiple scales[J]. International Journal of Computer Vision, 2020, 128(2): 438-459.
- [20] Li Shimiao. A review of feature detection and match algorithms for localization and mapping[J]. IOP Conference Series: Materials Science and Engineering, 2017, 231: 012003.
- [21] Dong J, Soatto S. Domain-size pooling in local descriptors: DSP-SIFT[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston: IEEE Press, 2015: 5097-5106.