

A New Image Feature Matching Algorithm Based on GPU Acceleration

Cong Li*, Dabo Guo, Xiaowen Liu

College of Physics and Electronic Engineering, Shanxi University, Taiyuan Shanxi
Email: *2013976644@qq.com

Received: Jan. 5th, 2019; accepted: Jan. 16th, 2019; published: Jan. 23rd, 2019

Abstract

Aiming at the problem that the traditional image matching algorithm has long matching time and high mismatch, this paper proposes a three-step feature extraction and matching algorithm (BRISK + FREAK + BF): Firstly, the feature points of the image are detected by the AGAST feature detection algorithm; then using the FREAK descriptor to describe the feature point neighborhood, assigning the main direction and recording gradient feature to the feature point; finally, using the BF algorithm to match the feature points of the image, and at the same time, using the GPU to speed up the algorithm. The improved algorithm is compared with the traditional BRISK algorithm and FREAK algorithm. It is found that the number of feature matching and the correct matching rate of image feature points are improved, and have the good robustness to scale difference and brightness difference of image, and running speed is increased by 35% - 40% compared to the faster BRISK algorithm.

Keywords

Feature Detection, Feature Extraction, BRISK, AGAST, FREAK, Image Processing

基于GPU加速的一种新图像特征匹配算法

李 聪*, 郭大波, 刘小文

山西大学, 物理电子工程学院, 山西 太原
Email: *2013976644@qq.com

收稿日期: 2019年1月5日; 录用日期: 2019年1月16日; 发布日期: 2019年1月23日

摘 要

针对传统图像匹配算法匹配时间较长、误匹配较高的问题, 本文提出了一种三步特征提取和匹配算法

*通讯作者。

(BRISK + FREAK + BF): 首先, 利用AGAST特征检测算法检测出图像的特征点; 然后用FREAK描述子对特征点邻域进行描述, 为特征点分配主方向和记录梯度特征; 最后使用BF算法对图像的特征点进行匹配。同时, 使用GPU对该算法进行加速处理。改进的算法经实验与传统的BRISK算法与FREAK算法比较分析后得知, 在图像的特征匹配数目及图像特征点匹配正确率上有了一定的提高, 对图像的尺度差异及亮度差异具有良好的鲁棒性, 且运行速度相较于较快的BRISK算法来说提高了35%~40%左右。

关键词

特征检测, 特征提取, BRISK, AGAST, FREAK, 图像处理

Copyright © 2019 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

在图像处理领域中, 图像匹配是一个研究热点, 它可以广泛应用于目标识别与跟踪[1]、遥感影像处理[2]等领域。

图像匹配通常分为三个步骤: 图像特征点的检测、特征点的提取及特征点的匹配。国内外学者对于图像匹配也有许多出色的研究, 例如 2004 年 David G. Lowe 整理完善所提出的 SIFT 算法[3], 它对于光线、噪声、微视角改变的容忍度较高; 2006 年 Herbert Bay 等人提出了 SURF 算法[4], 该算法的优点在于所提取的特征具有平移、缩放、旋转的不变性, 且相较于 SIFT 算法来说, 计算速度得到了提高; Calander 等人提出的 BRIEF 算法[5]和 Stefan 提出的 BRISK 算法[6]均是生成二进制链码描述符, 且都加速了特征向量匹配过程, 但是 BRIEF 描述符对于旋转非常敏感, 造成了其具有一定的局限性; 而 BRISK 算法相对而言较为稳定, 有更好的尺度与光照不变性, 且它的特征点的检测和描述子的计算比 BRIEF 的速度要快一个数量级; Alahi 等人于 2012 年提出了 FREAK 算法[7], 该算法是从人眼识别物体的原理中得到启发的, 它利用人类视网膜的特性来进行采样点的选择, 算法较为新颖, 但是该算法不具备尺度不变性, 在图像尺度发生变化的情况下, 特征点的匹配准确率会大大降低。

本文的主要工作有: 1) 使用 BRISK 算法对图像的特征点进行检测; 2) 用 FREAK 描述子对特征点进行描述; 3) 使用 BF 匹配算法对图像的特征点进行匹配; 4) 研究对比本文所提出的 BRISK + FREAK + BF 方法与传统的 BRISK 算法与 FREAK 算法, 比较分析后得知本文所提出的方法在特征点的数目及匹配的正确率上均有了一定的提高, 且在不同尺度与不同光照下具有良好的鲁棒性。同时, 使用 GPU 进行加速使本文算法的实时性也较高。

2. BRISK 算法

2.1. BRISK 算法

BRISK 算法是于 2011 年提出的特征提取算法, 它具有良好的旋转不变性、尺度不变性及较好的鲁棒性等。为实现尺度不变性, BRISK 算法首先应用高斯差分滤波器把图像构建成 N 组 S 层的高斯差分 DOG (Difference of Gaussian)金字塔[8], 如图 1 所示。高斯差分金字塔是由高斯金字塔构造得到的, 它是对同一个八度的两幅相邻的图像做差得到插值图像, 所有八度的这些插值图像的集合, 便构成了高斯差分金字塔。

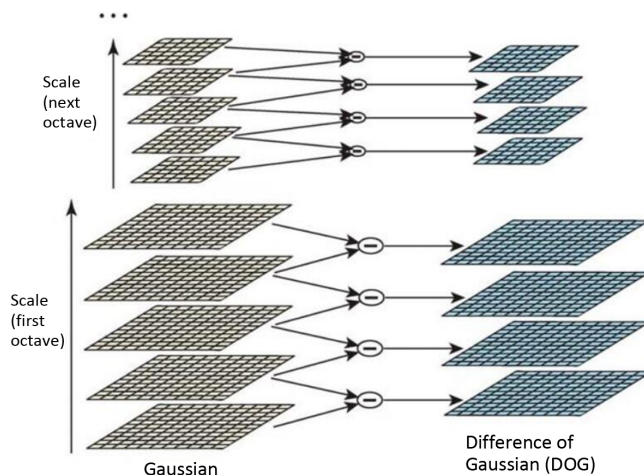


Figure 1. Gaussian difference pyramid
图 1. 高斯差分金字塔

DOG 金字塔的第一组第一层是由高斯金字塔的第一组第二层减去第一组第一层得到，第一组第二层是由高斯金字塔的第一组第三层减去第一组第二层得到的，依次类推，逐组逐层生成每一个差分图像，所有差分图像构成差分金字塔。高斯差分金字塔可用于后续的特征点的提取。

2.2. AGAST 特征检测

BRISK 算法中所采用的特征检测算法是 AGAST 算法，它可以提高尺度空间的准确性，AGAST 算法 [9] 是由 Mair 等人提出的一种改进的 FAST 算法 [10]，即自适应多尺度快速角点检测算法 AGAST (Adaptive Generic Accelerated Segment Test)。该算法在尺度不变的要求下，有着更高的计算效率和实时性。

AGAST 算法构建了更为精细的尺度空间金字塔，它由 n 个 octave 层 c_i 和 n 个中间层 d_i 组成 ($i = 0, 1, \dots, n-1$) [11]。 c_i 是通过对原始图像 c_0 的一次半采样得到的，中间层 d_0 是对 c_0 的 1.5 倍采样得到的，其余内层依次对 d_0 半采样得到，如图 2 所示。

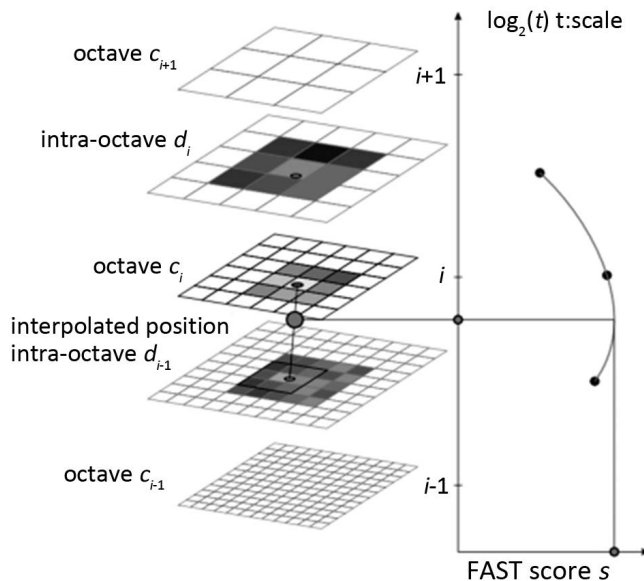


Figure 2. AGAST algorithm scale spatial feature point detection
图 2. AGAST 算法尺度空间特征点检测

AGAST 特征点检测算法主要分为以下两个步骤：第一，在每一层和内层分别运用阈值为 T 的 FAST9-16，即 16 个像素点中至少有连续 9 个像素点比中心像素点充分亮或者充分暗作为准则，来使用探测器检测潜在的兴趣域；第二，在尺度空间内对潜在的兴趣域中的各个像素点进行非极大值抑制；AGAST 特征检测算法为了能更加准确地定位特征点，对所检测出的每一个极大值都进行连续尺度优化及亚像素校正[12]。

3. FREAK 算法

3.1. FREAK 采样模式

生物学家研究表明，在人眼的视网膜区域中，感受光线的细胞的密度是不同的。根据感光细胞的密度将人眼的视网膜分为了 4 个区域：foveola、fovea、parafoveal、perifoveal [13]。其中，foveola 区域主要是对高分辨率的图像进行处理，perifoveal 主要是对低分辨率的图像进行处理，如图 3 所示。正是基于人类视网膜的结构，Alahi 等人才提出了 FREAK 算法。

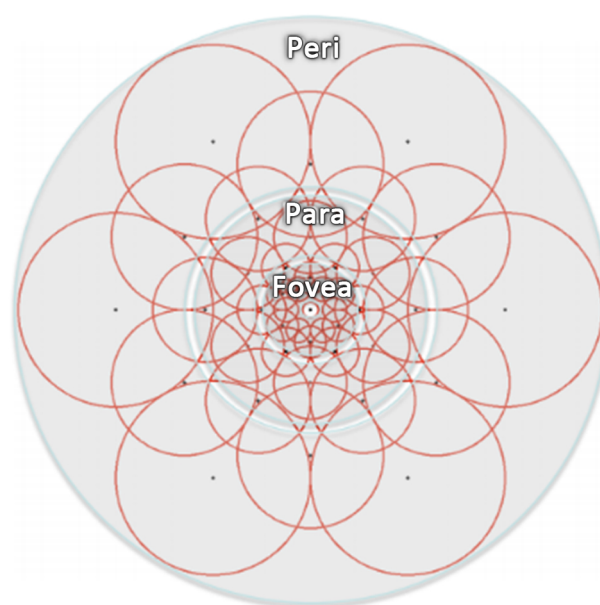


Figure 3. FREAK sampling mode
图 3. FREAK 采样模式

FREAK (Fast REtinA Keypoint)，即快速视网膜特征点提取算法，该算法采取了更接近人眼视觉系统接收图像信息的采样模式。该模式以特征点为中心，由 7 层同心圆环构成，每层圆环均匀取 6 个采样点。如果采样点离特征点的距离越近，采样点圆的半径越小，则采样点越密集；离特征点的距离越远，采样点圆的半径越大，则采样点越稀疏[14]。FREAK 算法对每个采样点进行了高斯平滑处理，每个圆圈的半径表示了高斯模糊的标准差。

3.2. FREAK 特征点描述子

FREAK 描述子由采样点对的强度比较结果级联组成，形成了二进制位串描述子[14]，用 F 表示，如下公式 1、2 所示：

$$F = \sum_{0 \leq a < N} 2^a T(P_a) \quad (1)$$

$$T(P_a) = \begin{cases} 1 & I(P_a^{r_1}) - I(P_a^{r_2}) > 0 \\ 0 & I(P_a^{r_1}) - I(P_a^{r_2}) \leq 0 \end{cases} \quad (2)$$

式中, P_a 表示当前采样点的位置, $I(P_a^{r_1})$ 表示采样点对 P_a 中前一个采样点的像素值, $I(P_a^{r_2})$ 表示后一个采样点的像素值。

假设共得到了 M 个采样点个数, 则描述向量共有 C_M^2 维生成, 但图像所包含的全部信息量并不全部存储于这些维度中, 有些维度是冗余的, FREAK 算法为了得到更具辨识度的描述子, 对所得到的描述子进行筛选, 保留信息量大的维度。

- 1) 首先构建一个矩阵 D , D 的每一行表示一个特征点二进制描述子, 每一行有 C_M^2 列;
- 2) 计算矩阵 D 的每一列的均值, 由于 D 中元素均是 0/1 分布的, 且为了得到独特性较好的特征, 所要求的方差应该较大, 所以取列的均值为 0.5;
- 3) 对所计算得到的每一列的方差进行由大到小的排序;
- 4) 最后选取前 512 列作为最终的二进制描述符, 也可以选择 256、1024 等, 但是发现只有前 512 列是最相关的, 再增加列数对结果并没有很大的提高。

上述筛选完成之后, FREAK 描述符周边的采样区域生成了低维度的描述向量, 而高维度的描述向量则生成在 FREAK 描述符中心的采样区域中。由此我们可以得到, FREAK 描述符周边的采样区域可以将目标图像的基本信息描述出来, 而目标图像的细节信息则会被 FREAK 描述符中心的采样区域所描述出来。

3.3. 为特征点分配方向

使用 FREAK 算法进行图像特征匹配之前, 还需要为所提取到的特征点分配主方向和记录梯度特征。由于特征点周围有 43 个采样点, 可产生 $43 \times (43 - 1) / 2 = 903$ 个采样点对, FREAK 算法选取其中 45 个长的、对称的采样点对来提取特征点的方向, 如下图 4 所示:

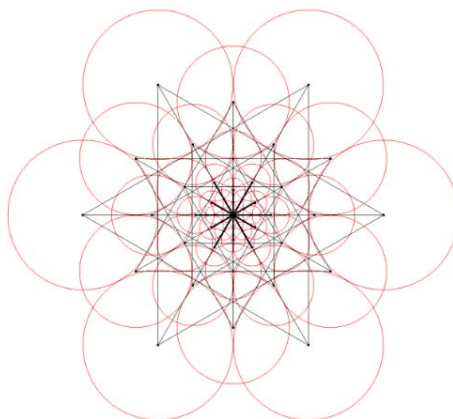


Figure 4. Sampling point pair
图 4. 采样点对

其特征点方向计算公式为:

$$O = \frac{1}{M} \sum_{P_o \in G} \left(I(P_o^{r_1}) - I(P_o^{r_2}) \right) \frac{P_o^{r_1} - P_o^{r_2}}{\|P_o^{r_1} - P_o^{r_2}\|} \quad (3)$$

其中, O 表示图像的局部梯度信息, M 表示采样点对的个数, G 表示采样点对的集合, P_o 表示采样点对的位置。

3.4. 特征点匹配

本文所采用的特征点匹配方法为 BF 算法，即暴力(Brute Force)算法[15]，它是一种普通的模式匹配算法，它的思想是将目标串 S 的第一个字符与模式串 T 的第一个字符进行匹配。若相等，则继续比较 S 的第二个字符和 T 的第二个字符；若不相等，则比较 S 的第二个字符和 T 的第一个字符，依次比较下去，直到得出最后的匹配结果。

BF 算法应用于图像特征点匹配领域中，首先在第二幅图像中选取一个特征点然后依次与第一幅图像的每个关键点进行距离测试，最后返回距离最近的特征点。BF 算法较为直接，是一种蛮力算法，时间复杂度为 $O(M*N)$ ，虽牺牲了时间但换取了该算法的准确性，所以本文采用该匹配方法。

4. GPU 加速

GPU (Graphics Processing Unit) [16]即图形处理器，又称显示核心、视觉处理器、显示芯片，它可以工作于多种场合，例如个人电脑、游戏机等。由于在现代的计算机中图形处理变得越来越重要，所以需要有一个专门的用于图形处理的核心处理器，这时候 GPU 便显示出它优越的性能，GPU 可以进行并行计算，显著提高了图形处理的实时性。它是显示卡的“心脏”，相当于是专用于图形处理的 CPU，在图像处理领域中它更专业更强大，且它的工作效率要远高于 CPU。

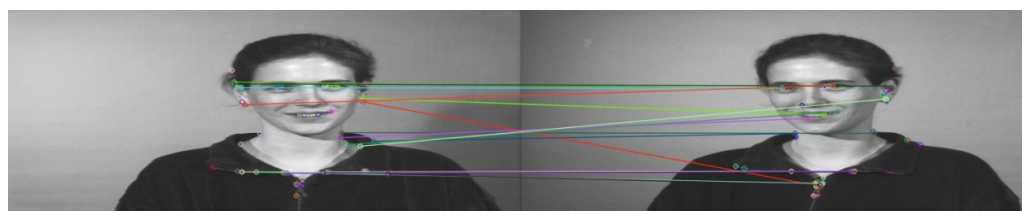
NVIDIA 提供了调用 GPU 的 CUDA 核心的接口，CUDA 是一个并行计算架构，它为 GPU 并行提供了一个平台，并利用虚拟指令来访问并行计算资源，可以使大量程序员能运用所学语言进行编程。GPU 适合较大规模的数据计算，相比 CPU 来说，能大大缩短运行时间，提高实时性。所以本实验在 Visual Studio2015 的基础上安装了 CUDA9.0 并配置了环境变量，确保了实验的顺利进行。

5. 实验及结果

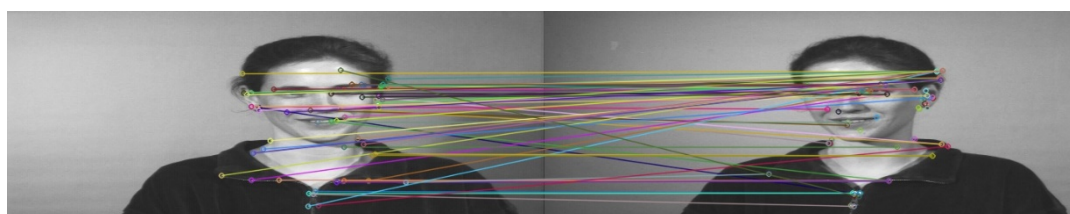
本实验平台是基于 windows7 + Visual Studio2015 + Opencv3.3.1 + CUDA9.0，所采用的测试图像为 anne、michel，其像素分别为 720×288 、 320×240 ，图像匹配结果如下所示。本文实验首先使用 BRISK 特征检测算法检测出图像的关键点，其次使用 FREAK 算法提取出图像的特征点，最后利用 BF 匹配算法对所提取到的特征点进行匹配，得到图像的匹配结果。

5.1. 尺度不同对比实验

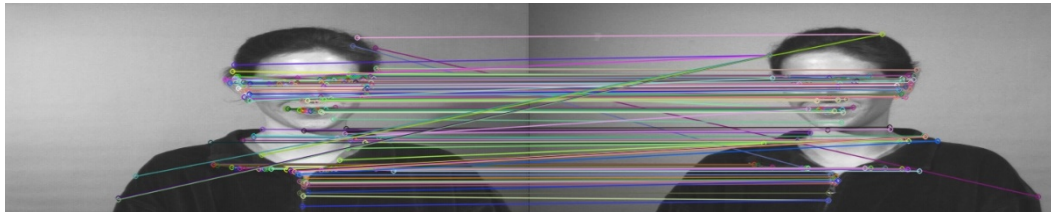
对于尺度变化的图像来说，例如图像 anne，所得到的图像匹配结果如下图 5 所示：



BRISK 算法



FREAK 算法



本文算法

Figure 5. Matching image for each algorithm of anne
图 5. Anne 的各算法的匹配图像

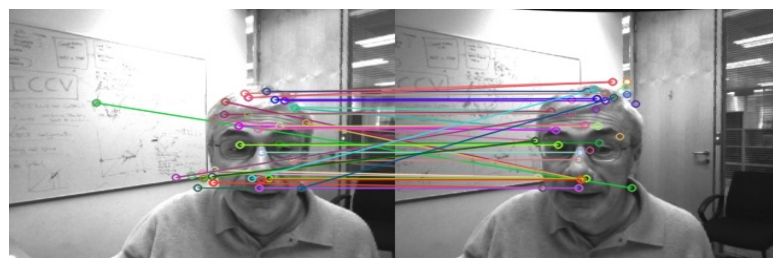
Table 1. Match results of each algorithm of anne
表 1. Anne 的各算法的的匹配结果

anne	匹配数目	正确个数	错误个数	正确率	错误率	运行时间(ms)
BRISK	16	9	7	56.25%	43.75%	18
FREAK	58	22	36	37.93%	62.07%	31
本文算法	165	150	15	90.91%	9.09%	11

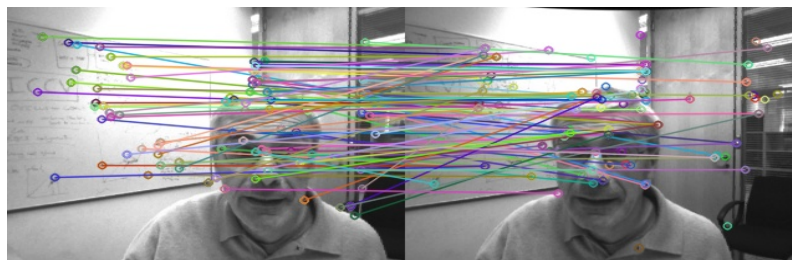
表 1 通过对不同尺度的图像的匹配结果进行分析, 我们发现 BRISK 算法与 FREAK 算法所提取到的特征点较少且匹配的正确率不高, 由于 FREAK 算法不具备尺度不变性, 可以发现 FREAK 算法虽提取到的特征点较 BRISK 算法来说较多但正确率却有所降低。而本文的算法考虑到了 BRISK 特征检测的尺度不变性, 与 FREAK 描述子进行结合所得到的匹配结果具有一定的尺度不变性。本文算法与 FREAK 算法相比来说, 匹配数目有了近 3 倍的提升; 与匹配的正确率较高的 BRISK 算法相比来说, 本文算法的正确率也提高到了 90.91%, 可以看出本文算法在对图像有一定的尺度差异的情况下具有良好的鲁棒性。从程序的运行时间来看, 本文算法使用了 GPU 并行处理, 所用时间比 BRISK 和 FREAK 两种算法都少, 并且相较于速度较快的 BRISK 算法来说, 本文算法的运行速度提升了 38.88%。

5.2. 亮度不同对比实验

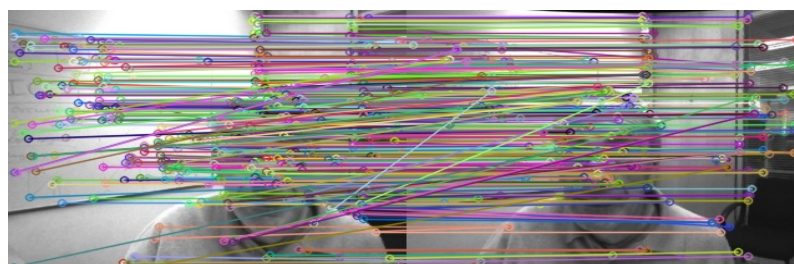
对于光照变化的图像来说, 例如图像 Michel, 所得到的图像匹配结果如下图 6 所示:



BRISK 算法



FREAK 算法



本文算法

Figure 6. Matching image for each algorithm of Michel
图 6. Michel 的各算法的匹配图像

Table 2. Match results of each algorithm of Michel
表 2. Michel 的各算法的匹配结果

Michel	匹配数目	正确个数	错误个数	正确率	错误率	运行时间(ms)
BRISK	35	21	14	60%	40%	23
FREAK	85	29	56	34.12%	65.88%	31
本文算法	407	376	31	92.38%	7.62%	14

表 2 通过对不同光照条件下的图像的匹配结果进行分析, 仍然也可以发现, BRISK 算法所提取到的特征点较少, 但匹配的正确率却比 FREAK 算法高; FREAK 算法所提取的特征点较多但正确率却有所降低, 这是由于 FREAK 算法不具有光照不变性, 本文算法较其余两种方法来说特征匹配数目及正确率均有所提高。本文算法与 FREAK 算法相比来说, 匹配数目有了近 5 倍的提升, 且本文算法的正确率较 BRISK 来说提高到了 92.38%, 所以本文算法在对图像有一定的光照差异的情况下也具有较好的鲁棒性。同时, 经过 GPU 的加速, 本文算法的运行时间仅为 14 ms, 相比 BRISK 算法来说提高了 39.13%。

6. 结论

通过上述表述, 我们对比了 BRISK 算法、FREAK 算法以及本文提出的 BRISK + FREAK + BF 算法, 并且研究了本文算法在不同尺度及不同光照条件下的图像匹配结果, 研究结果表明, 本文算法无论是在尺度不变性还是光照不变性上都优于这两种算法。BRISK 算法优点在于其运行速度很快, 但同时该算法所提取到的特征点也较少; FREAK 算法由于不具备尺度不变性与光照不变性, 所以其匹配准确度过低, 匹配结果误匹配较多。而本文提出的算法则很好地结合了这两种算法的优点, 本文只选取了两组较具有代表性的图像结果, 但经过大量的实验图像数据表明, 本文算法的特征匹配的数目较 FREAK 算法来说提高了 3~5 倍, 匹配的正确率较 BRISK 算法来说从 50%~60%提高到了 90%左右, 且本文算法运行于 GPU 加速平台上, 相比 BRISK 算法来说, 速度提高了 35%~40%左右。可以证明本文算法对图像的尺度差异及光照差异具有良好的鲁棒性, 且实时性有了较大的提高。同时, 本文也存在不足的地方, 如何进一步提高算法的匹配准确率及实时性将是未来的研究方向。

参考文献

- [1] 吕娜, 冯祖仁. 图像匹配与跟踪研究[J]. 西安交通大学学报, 2010, 44(10): 23.
- [2] 余先川, 吕中华, 胡丹. 遥感图像配准技术综述[J]. 光学精密工程, 2013, 21(11): 2960-2972.
- [3] Lowe, D.G. (2004) Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, **60**, 91-110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- [4] Bay, H., Tuytelaars, T. and Gool, L.V. (2006) SURF: Speeded up Robust Features. *European Conference on Computer*

- Vision*, **3951**, 404-417. https://doi.org/10.1007/11744023_32
- [5] Calonder, M., Lepetit, V., Strecha, C. and Fua, P. (2010) BRIEF: Binary Robust Independent Elementary Features. *European Conference on Computer Vision*, **6314**, 778-792. https://doi.org/10.1007/978-3-642-15561-1_56
- [6] Leutenegger, S., Chli, M. and Siegwart, R.Y. (2011) BRISK: Binary Robust Invariant Scalable Keypoints. *International Conference on Computer Vision*, **58**, 2548-2555.
- [7] Alahi, A., Ortiz, R. and Vandergheynst, P. (2012) FREAK: Fast Retina Keypoint. *Computer Vision and Pattern Recognition*, Providence, RI, 16-21 June 2012, 510-517.
- [8] 石祥滨, 孙奇, 张德园, 刘芳. 一种自适应阈值分块 BRISK 的图像配准方法[J]. 沈阳航空航天大学学报, 2014, 31(3): 65-72.
- [9] Mair, E., Hager, G.D., Burschka, D., Suppa, M. and Hirzinger, G. (2010) Adaptive and Generic Corner Detection Based on the Accelerated Segment Test. Springer, Berlin Heidelberg.
- [10] 周莉莉, 姜枫. 基于 FAST 和 BRIEF 的图像匹配算法[J]. 计算机工程与设计, 2015(5): 1269-1273.
- [11] 龙思源, 张葆, 宋策, 孙保基. 基于改进的加速鲁棒特征的目标识别[J]. 中国光学, 2017, 10(6): 719-725.
- [12] 张恒, 刘大勇, 刘艳丽, 聂晨曦. 融合深度信息的 BRISK 改进算法[J]. 计算机应用, 2015, 35(8): 2285-2290.
- [13] 王灿进, 孙涛, 陈娟. 基于 FREAK 特征的快速景象匹配[J]. 电子测量与仪器学报, 2015(2): 204-212.
- [14] 叶志坚, 王福龙. 一种改进的 FREAK 算法的图像特征点匹配[J]. 广东工业大学学报, 2017(6): 37-42.
- [15] 郑盼盼. 数据结构 Brute-Force 算法的实现[J]. 电子制作, 2014(6x): 85-86.
- [16] 李刚. 基于 GPU 加速的人脸识别实现[J]. 电脑知识与技术, 2018, 14(12): 185-187.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2161-8801, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>
期刊邮箱: csa@hanspub.org