

一种基于最佳伙伴相似性的快速图像匹配算法

吕波凯*, 吴成茂, 田小平

西安邮电大学电子工程学院, 陕西 西安 710121

摘要 针对最佳伙伴相似性(BBS)图像匹配算法计算复杂度高、目标定位不准确等问题,提出了一种改进的图像匹配算法。根据模板图像的尺寸自适应选择图像分块大小,以减少匹配点集中点的数目,从而减小 BBS 算法的运算量;根据子块的灰度值重新排列子块,在此基础上得到 BBS 的置信度图,从置信度图中筛选出目标的可能位置,并重新计算目标可能位置的真实 BBS 分数;用目标可能位置的真实 BBS 分数替换通过双线性插值得到的 BBS 分数,将目标可能位置中 BBS 分数最高的位置作为匹配结果。实验结果表明,该算法可降低 BBS 算法的运行时间,同时提高目标定位的准确度。

关键词 模板匹配; 最佳伙伴相似性; 自适应分块; 置信度图; 双线性插值

中图分类号 O436

文献标志码 A

doi: 10.3788/LOP57.101018

Fast Image Matching Algorithm Based on Best-Buddies Similarity

Lü Bokai*, Wu Chengmao, Tian Xiaoping

School of Electronic Engineering, Xi'an University of Posts & Telecommunications, Xi'an, Shaanxi 710121, China

Abstract An improved image matching algorithm is proposed to solve the problems of high computational complexity and inaccurate target positioning of best-buddies similarity (BBS) image matching algorithm. According to the size of the template image, the size of image blocks is correspondingly selected to reduce the number of points in the matching point set, and then to reduce the computation of the BBS algorithm. The sub blocks are rearranged according to their gray values, and thus the BBS confidence map of is obtained. The possible location of the target is screened out from the confidence map, and the true BBS score of the possible position of the target is recalculated. The BBS score obtained by bilinear interpolation is replaced by the real BBS score of the recalculated possible location of the target. The location with the highest BBS score among the possible locations is taken as the matching result. Experimental results show that the algorithm improves the accuracy of the target positioning while reducing the running time of the BBS algorithm.

Key words template matching; best-buddies similarity; adaptive block; confidence map; bilinear interpolation

OCIS codes 100.2960; 100.2000; 100.4994; 100.3008

1 引言

图像匹配是视觉应用的核心问题^[1],广泛应用于日常生活中^[2],如机器人自主行驶^[3]、视觉导航^[4]、特征提取^[5]、立体视觉匹配^[6]领域。模板匹配算法是在目标图像上逐像素滑动搜索窗口找到与所选模板图像内容最接近的区域^[7],随着图像匹配技术的发展,国内外学者提出了多种图像匹配算法^[8]。但仍存在许多不足,在目标变化复杂时难以有效匹

配,如背景变化、物体形变、遮挡变化^[9]。

现有的模板匹配算法主要分为基于像素灰度值本身和基于几何特征的匹配算法,基于像素灰度值本身的匹配算法,原理简单容易实现^[10]。如归一化互相关(NCC)模板匹配算法,对有噪声干扰和亮度变化的图像稳健性较好^[11],但计算复杂度高、匹配耗时长。基于几何特征的匹配算法,利用提取出的模板图像特征,如边缘、颜色、角点,计算相似性,对目标旋转、尺度变化不敏感,在动态情况下匹配效果

收稿日期: 2019-09-02; 修回日期: 2019-10-16; 录用日期: 2019-10-25

基金项目: 国家自然科学基金(61671377,51709228)、陕西省自然科学基金(2017JM6107)

* E-mail: 976650565@qq.com

较好^[12],但该算法结构复杂、计算量较大。

Dekel等^[13]提出的模板匹配算法——最佳伙伴相似性(BBS)图像匹配算法,将模板图像和搜索窗口看作两个点集,利用点集间的相似点对数量描述图像的相似性,能有效匹配背景杂乱、目标变形剧烈的图片。相比统计直方图^[14]、归一化互相关^[15]和误差平方和(SSD)^[16]等算法,该算法匹配精度更高、稳健性更好。文献[17]将BBS的置信度图经筛选后进行均值滤波,同时考虑了搜索窗口周围区域的信息,提高了算法在目标发生部分遮挡时的稳定性。文献[18]使用多尺度组合分组(MCG)算法^[19]在待匹配图像中生成多个目标可能位置(Proposal),通过上采样使模板和Proposal的尺寸统一为两者中较大者,然后计算模板图像和所有Proposal的BBS分数,增强了算法应对目标尺度变化的能力。

文献[17]、[18]中对BBS算法的改进没有解决运算量大、目标定位不准确的问题。原始BBS算法为了减小运算量,将模板图像和待匹配图像分割为互不重叠的子块,将每个子块视为一个点,搜索窗口在待匹配图像上滑动相当于以子块大小为步长遍历待匹配图像。未被遍历位置的BBS分数(相似性度量)通过双线性插值得到,但并不是该位置的真实BBS分数,所以确定的目标位置并不准确。本文根据模板图像的尺寸自适应选择图像分块尺寸,以减小匹配点集中点的数目,从而减小运算量。先计算模板图像和待匹配图像的BBS分数,从得到的置信度图中筛选出目标可能位置,并重新计算这些位置的真实BBS分数,选取BBS分数最高的位置作为匹配结果,从而解决双线性插值带来的目标定位不准确问题。

2 基本原理

传统模板匹配算法需要计算模板图像和待匹配图像中所有点的相似性,当模板图像中含有背景或大量异常值(如遮挡、噪声、光照不均匀)时,会影响算法的稳定性,导致匹配精度下降。而BBS算法只考虑模板图像和待匹配图像之间的最佳伙伴点对,相比传统模板匹配算法稳健性更强。

2.1 最佳伙伴相似性

BBS算法将模板图像和待匹配图像看作两个点集,利用两点集间的最佳伙伴点对数量描述模板图像和待匹配图像的相似性。存在点集 $P = \{p_i\}_{i=1}^N, Q = \{q_j\}_{j=1}^M$,其中 M, N 分别为点集 P, Q 包含的点数目, $p_i, q_j \in R^d$,其中 R^d 为 d 维空间中

的点,当 p_i, q_j 互为最近邻时, p_i 和 q_j 为一对最佳伙伴点对(BBP),可表示为

$$b(p_i, q_j, P, Q) = \begin{cases} 1, & L(p_i, Q) = q_j \ \& \ L(q_j, P) = p_i \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

式中, $\&$ 为与运算, $L(p_i, Q) = \arg \min d(p_i, q_j)$ 为点 p_i 到 Q 的最小距离。BBS的距离度量为两点间的灰度值差异以及空间位置差异的线性加权,点 p_i 到 q_j 的距离可表示为

$$d(p_i, q_j) = |p_i^{(A)}, q_j^{(A)}|_2^2 + \lambda |p_i^{(K)}, q_j^{(K)}|_2^2, \quad (2)$$

式中, A 为对应点的灰度值, K 为对应点的空间位置, λ 为空间距离权重,实验设置为2。归一化点集 P, Q 间的BBP数目,就能得到点集间的相似性, P, Q 间的BBS分数可表示为

$$X_{\text{BBS}}(P, Q) = \frac{1}{\min\{M, N\}} \sum_{i=1}^N \sum_{j=1}^M b(p_i, q_j, P, Q), \quad (3)$$

式中, $X_{\text{BBS}}(P, Q)$ 使用 M, N 中较小的归一化BBP数量。BBS算法采用滑动搜索框在待匹配图像上寻找目标,搜索框尺寸与模板图像尺寸相同,即 $M = N$ 。

2.2 粗匹配

BBS算法采用滑动搜索框在待匹配图像上寻找目标,计算搜索框与模板图像的BBS相似性,遍历待匹配图像上每个像素点,得到BBS算法的置信度图。假定待匹配图像包含 U 个像素点、模板图像包含 Y 个像素点,BBS算法需要计算模板图像中所有像素点与待匹配图像中每个像素点的距离,计算复杂度为 $O(dUY)$ 。为了减小运算量,原始BBS算法将模板图像和目标图像分割成尺寸为 $3 \text{ pixel} \times 3 \text{ pixel}$ 且互不重叠的子块,然后将每个子块看作一个点,构成点集再计算相似性。若将图像分割成尺寸为 $s \times s$ 、且互不重叠的子块,则待匹配图像、模板图像包含的点数分别为 $U/s^2, Y/s^2$,点的计算复杂度为分块前的 s^2 倍,分块后算法的计算复杂度可表示为

$$O(s^2 d \frac{U}{s^2} \frac{Y}{s^2}) = O(d \frac{UY}{s^2}). \quad (4)$$

由(4)式可知,当模板图像和待匹配图像的尺寸确定时,BBS算法的计算复杂度与 s^2 成反比,增大 s 可以减小计算复杂度。但随着 s 的增大,模板图像和待匹配图像分成的子块数目随之减少,匹配点集中的点的数目也减少了。为保证算法成功匹配到目标区域,模板图像中的点集必须包含足够多的点。实际应用中的模板图像尺寸差距较大,若采用相同的 s ,

将尺寸较大的目标分成的子块数目过多,导致运算量过大;将尺寸较小的目标分成的子块数目过少,导致小尺寸目标的匹配精度下降甚至匹配到背景区域,因此需要根据目标图像的尺寸自适应调整 s 。

根据模板图像包含的像素数目将目标分类,对不同尺寸的目标采用不同的 s ,具体的分类标准如表 1 所示。

表 1 不同尺寸目标选择的 s

Table 1 Choice of s for different size targets

Number of template pixels	0–2000	2000–5000	5000–10000	10000–20000
s /pixel	4	5	6	7

可以发现,改进后的算法根据模板图像的尺寸

选择 s ,即大尺寸模板选择大尺寸分块,小尺寸模板选择小尺寸分块,模板图像像素数目每增加一倍分块尺寸 s 加 1 pixel。

将图像分割后的子块看作一个点,搜索框在待匹配图像上每次滑动一个点,相当于以子块尺寸为步长在图像上搜索目标。如子块尺寸为 $3 \text{ pixel} \times 3 \text{ pixel}$,则搜索框以 3 pixel 为步长在待匹配图像上滑动,未被遍历位置的 BBS 分数通过双线性插值得到,得到所有位置的 BBS 分数,由全部位置的 BBS 分数构成 BBS 置信度图,如图 1 所示。其中,图 1 (a)为计算得到的真实 BBS 分数,两个相邻 BBS 分数的间距 $s = 3 \text{ pixel}$,图 1(b)为双线性插值得到的最终 BBS 置信度图。

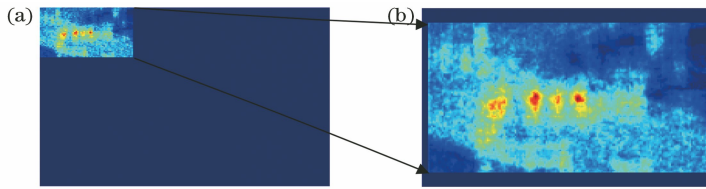


图 1 原始 BBS 算法的置信度图。(a)计算得到的真实 BBS 分数;(b)双线性插值得到的最终 BBS 置信度图

Fig. 1 Confidence map of original BBS algorithm. (a) Real BBS score by calculation; (b) final confidence map obtained by bilinear interpolation

如图 1(a)所示,以 s 为步长得到的置信度图仅为待匹配图像的 $1/s^2$,其余位置的 BBS 分数通过双线性插值得到,即双线性插值点相邻 4 个点 BBS 分数的线性加权。BBS 算法将置信度图中 BBS 分数最大的位置作为匹配结果,而双线性插值点的 BBS 分数不超过相邻 4 个点的最大值,若目标在双线性插值区域会导致目标定位不准确,如 $s = 3 \text{ pixel}$ 时,真实值区域仅为待匹配图像的 $1/9$,目标定位不准

确会导致匹配精度下降。若目标位置相邻 4 个点的 BBS 分数有大于一个最大值时,双线性插值会产生多个最大值,原始 BBS 算法并未给出 BBS 分数中出现多个最大值的解决办法,重新计算目标可能位置的 BBS 分数可以避免这种情况。双线性插值导致出现多个最大值的示意图如图 2 所示。图 2(a)中的矩形框为模板图像,图 2(b)中的矩形框为待匹配图像,图 2(c)为 BBS 算法得到的置信度图,图 2(d)

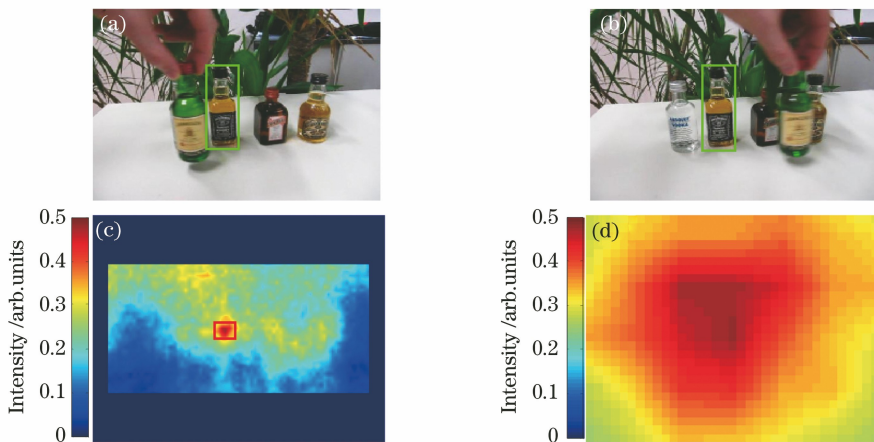


图 2 双线性插值导致多个最大值。(a)矩形区域为模板图像;(b)待匹配图像;(c) BBS 置信度图;(d)图(c)中矩形区域的放大图

Fig. 2 Bilinear interpolation leads to multiple maxima. (a) Template image marked in rectangle; (b) target image; (c) confidence map of BBS; (d) enlarged view of rectangular area in figure (c)

为图 2(c)中矩形区域的放大图。可以发现,随 s 的增大,目标位于双线性插值区域的概率会越大,匹配结果相对于目标真实位置的偏差也越大,最终导致匹配精度降低。

此外,图像分成的子块本身具有一定结构,当目标发生旋转、形变时不能保证子块的稳定性,随着 s 的增大,子块受到目标自身形变的影响也会增大。为了增强子块的稳定性,使用像素灰度值重新排列子块,原理如图 3 所示。图 3 中左侧为原子块,右侧为重新排列后的子块。若匹配图像为 RGB(Red, Green, Bule)图像,则使用 RGB 三分量的均值重新排列子块。

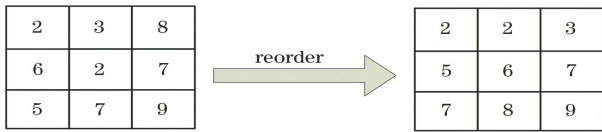


图 3 重新排列子块

Fig. 3 Rearranged subblocks

2.3 目标精确定位

为了解决原始 BBS 算法目标定位不准确的问题,从粗匹配得到的最终置信度图中筛选出目标可能出现的位置(BBS 分数高的位置),然后计算得到每个位置的真实 BBS 分数,取 BBS 分数最高的位置作为匹配结果,目标精确定位过程各阶段的置信度图如图 4 所示。图 4(a)为图 2 中模板图像和待匹配图像的置信度图,图 4(b)为筛选出的目标可能位置,图 4(c)为目标可能位置重新计算的 BBS 分数,图 4(d)为图 4(c)中矩形区域的放大图。

从图 4(d)中可以看出,重新计算的 BBS 分数构成的置信度图中只有一个极大值,并且是目标可能位置的真实 BBS 分数。因此从图 4(c)的置信度图中确定目标位置,可以解决原始 BBS 算法目标定位不准确的问题。筛选目标可能位置时,将粗匹配得到的置信度图中所有 BBS 分数进行排序,只保留分数靠前的位置,不同尺寸模板需要筛选的目标可能位置的数量 n 如表 2 所示。

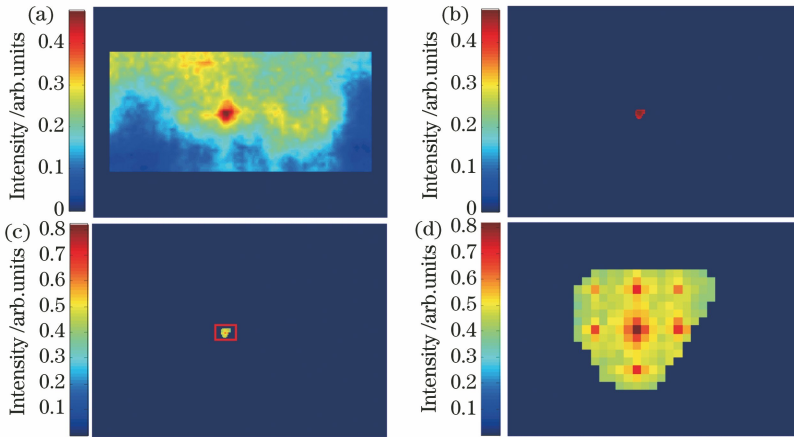


图 4 目标精确定位过程的 BBS 置信度图。(a)原始 BBS 的置信度图;(b)筛选出的目标可能位置;(c)目标可能位置重新计算的 BBS 分数;(d)图(c)中矩形区域的放大图

Fig. 4 Confidence map of target precise positioning. (a) Confidence map of original BBS; (b) possible location of the target; (c) BBS score of recalculated possible location; (d) enlarged view of rectangular area in figure (c)

从 1 开始逐渐增大 n ,算法的匹配精度逐渐增加,当 n 达到表 2 给出的数值后,算法的匹配精度无法继续提升。需要筛选的目标可能位置的数目最多为 150,只需要少量计算即可完成对目标的精确定位。总体上,尺寸越小的目标受到 s 增大的影响越大,需要筛选更多的目标可能位置完成目标定位。重新计算目标可能位置的 BBS 分数时,模板图像和待匹配图像的分块尺寸 s_1 同样根据模板尺寸自适应选取,实验中选择的 $s_1 = s - 1$ pixel。如图 3 中的模板图像尺寸为 156 pixel \times 54 pixel,即 8424 个像素,选择 s 为 6 pixel,则 s_1 为 5 pixel,需要筛选的目

标可能位置的数目为 80。

表 2 目标可能位置的数量

Table 2 Number of possible location

Number of template pixels	0–2000	2000–5000	5000–10000	10000–20000
n	150	80	80	50

本算法主要分为图像预处理、粗匹配、目标精确定位三部分。1)图像预处理:根据输入的模板图像尺寸自适应选择分块尺寸 s ,将输入的模板图像和待匹配图像分割成尺寸为 $s \times s$ 、且互不重叠的子块,若模板图像和待匹配图像尺寸不能被 s 整除,则

通过丢弃部分图像调整图像尺寸,最后使用像素灰度值重新排列子块。2)粗匹配:计算模板图像子块与当前搜索窗口子块的距离,根据求出的距离确定当前搜索窗口与模板图像的最佳伙伴点对的数目,归一化最佳伙伴点对数目得到当前位置的BBS分数。遍历待匹配图像,通过双线性插值得到未被遍历位置的BBS分数,由所有位置的BBS

分数构成粗匹配的置信度图。3)目标精确定位:从粗匹配得到的置信度图中筛选出一定数目的目标可能位置,然后计算每个目标可能位置的BBS分数,得到所有目标可能位置的真实BBS分数,计算时 $s_1 = s - 1$ pixel。最终将目标可能位置中BBS分数最大的位置作为匹配结果输出,算法流程图如图5所示。

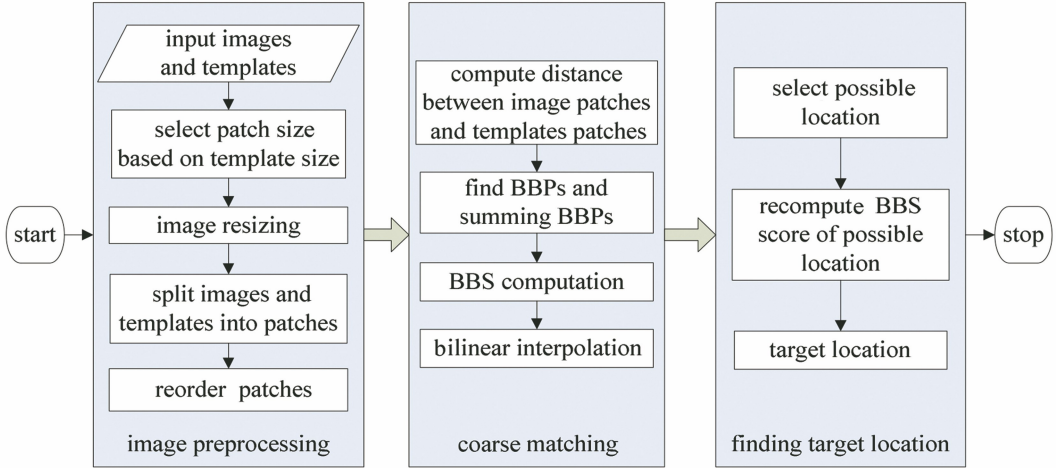


图5 算法流程图

Fig. 5 Flow chart of algorithm

3 实验分析与讨论

采用与原始BBS算法相同的数据集作为测试数据,测试数据由Wu等^[20]制作的标准视频序列衍生而来,标准视频序列共35段彩色视频,从每段视频随机选取3组图像对,共生成105组测试图像对,每组图像(模板图像和待匹配图像)的间隔为20 frame。在相同环境下从匹配速度和匹配精度两方面对本算法和原始BBS算法进行比较,测试环境: Intel Core i3 3110 m, 内存10 GB, 软件为 Matlab R2013a。

3.1 匹配速度比较

原始BBS算法匹配105组图像所需时间为6500.0 s, 每组图像约为61.9 s; 本算法匹配105组图像所需时间为494.0 s, 每组图像约为4.7 s, 相比原始BBS算法匹配速度有明显提升, 仅为原始BBS算法的7.6%。

为了适应不同尺寸的目标, 根据目标(模板)图像的尺寸将目标分类, 测试图像中各类目标的数量如表3所示。

由(4)式可知, BBS算法的运算量与分块尺寸 s 成反比, s 越大匹配速度越快, 为了直观了解本算法对各尺寸目标匹配时的速度, 分别统计了本算法和

表3 各尺寸目标的数目

Table 3 Number of different size templates

Number of template pixels	0-2000	2000-5000	5000-10000	10000-20000
Number of templates	41	48	10	6

原始BBS算法匹配各类目标的总耗时以及平均耗时, 结果如表4所示。

表4 两种算法的匹配时间

Table 4 Matching time of the two algorithms

Template	Total time / s		Average time / s	
	BBS	Proposed	BBS	Proposed
0-2000	222.3	135.0	5.4	3.3
2000-5000	1546.8	184.4	32.2	3.9
5000-10000	2180.1	99.5	218.0	10.0
10000-20000	2621.2	75.2	436.9	12.5

3.2 匹配精度以及匹配结果对比

一般采用模板图像和目标的重叠率衡量模板匹配算法的匹配精度, 重叠率 $A = \frac{T_c \cap T_b}{T_c \cup T_b}$, 其中 T_c 为匹配的真值区域, 由人工标记, T_b 为算法得到的匹配区域, \cap 表示交集, \cup 表示并集, 可以看出 A 越大匹配精度越高。依次计算每一对图像的匹配重叠率, 统计重叠率大于某个阈值(一般取0到1之间, 实验中阈值步长为0.05)的图像对数目占总图像

对数目的比例,得到算法的成功率曲线。算法的成功率曲线越靠近 1 匹配精度越高,成功率曲线与坐标轴围成多边形的面积叫作成功率曲线下面积(AUC),AUC 越大,算法的平均匹配精度就越高。本算法与原始 BBS 算法的成功率曲线及 AUC 如图 6 所示。

从图 6 中可以看出,改进算法的 AUC 相比于原始 BBS 的 AUC 提高了 0.032,这表明相比原始 BBS 算法本算法的匹配精度更高;在重叠率大于 0.6 时,本算法的成功率明显优于原始 BBS 算法,这证明了本算法对目标的定位更准确。

从每类图像中选两组图像的匹配结果进行比较,本算法和原始 BBS 算法的匹配结果如图 7 所示,图 7(a)为模板图像,图 7(b)为待匹配图像,其中

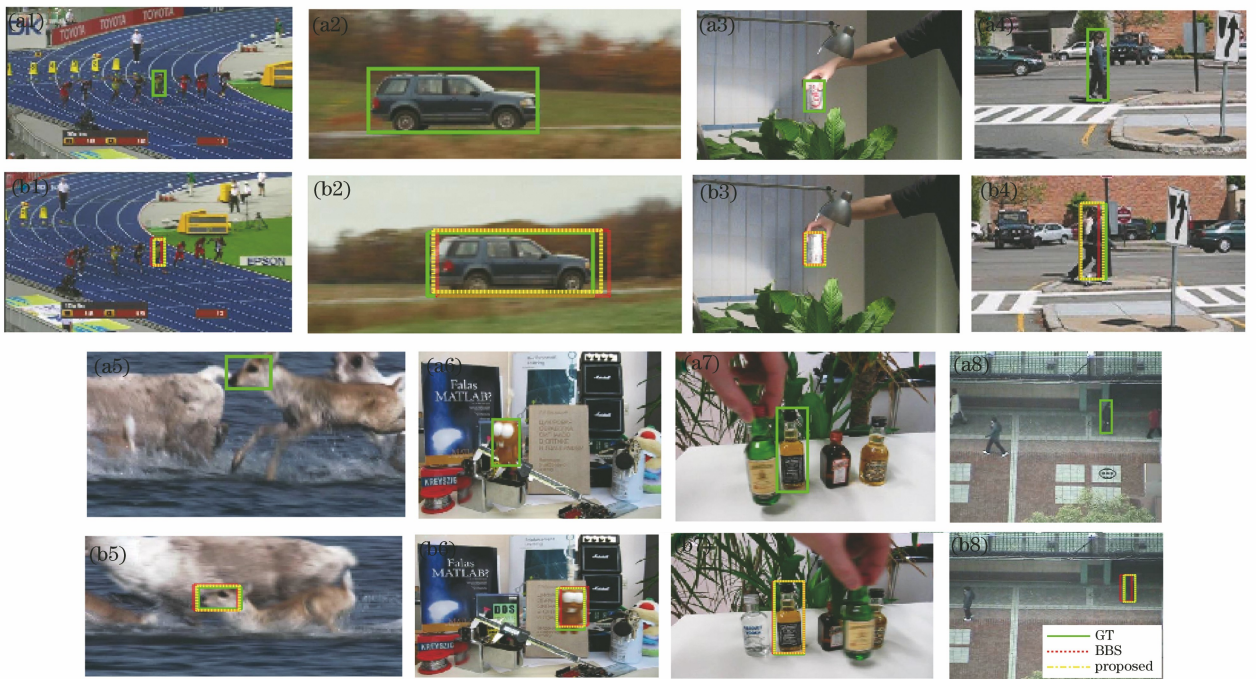


图 7 两种算法的匹配结果。(a)模板图像;(b)匹配结果

Fig. 7 Matching results of two algorithms. (a) Template images; (b) matching results

4 结 论

为了减小 BBS 算法的运算量,对不同尺寸的目标选取合适的分块大小,减少了匹配点集中点的数量,从而减小 BBS 算法的计算复杂度,提高了算法的计算速度。将图像分成的子块按照灰度值大小重新排列,增强了子块的稳定性。通过筛选出目标可能位置并计算这些位置的真实 BBS 分数,取最大值为目标位置,解决了原始 BBS 算法由双线性插值导致的目标定位不准确问题。实验结果表明,该算法比原始 BBS 算法的匹配速度更快,目标定位更准确。但 BBS

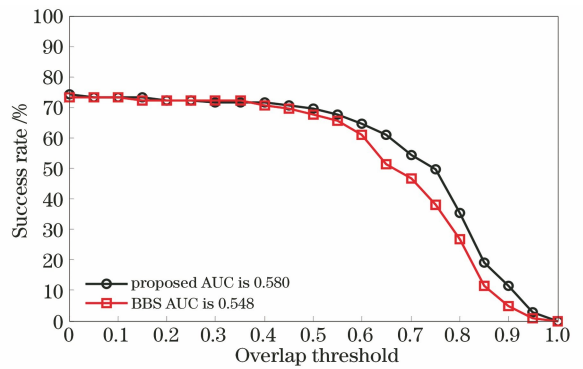


图 6 不同算法的成功率

Fig. 6 Success rate of different algorithms

GT 表示真值。可以看出,本算法和原始 BBS 算法均成功匹配到了目标区域,但本算法对目标的定位更准确,且匹配结果与目标真值的重叠率更高。

算法在目标对比度低时的匹配成功率较低,不能有效区分目标区域和背景区域,还需进一步改进。

参 考 文 献

- [1] Zhao P T, Da F P. Image matching with large viewing angle based on local features[J]. Acta Optica Sinica, 2019, 39(5): 0510002.
赵鹏图, 达飞鹏. 基于局部特征的大视角图像匹配[J]. 光学学报, 2019, 39(5): 0510002.
- [2] Jia D, Zhu N D, Yang N H, et al. Image matching methods[J]. Journal of Image and Graphics, 2019, 24(5): 677-699.

- 贾迪, 朱宁丹, 杨宁华, 等. 图像匹配方法研究综述[J]. 中国图象图形学报, 2019, 24(5): 677-699.
- [3] Warrant E, Dacke M. Visual navigation in nocturnal insects[J]. *Physiology*, 2016, 31(3): 182-192.
- [4] Zhang Q P, Cao Y. Image matching algorithm based on exposure and color information [J]. *Laser & Optoelectronics Progress*, 2019, 56(19): 191004 .
张庆鹏, 曹宇. 一种融合光照和彩色信息的图像匹配算法[J]. 激光与光电子学进展, 2019, 56(19): 191004.
- [5] Zhao S, Wang B, Tang C Y. Armvein feature extraction and matching based on chain code[J]. *Acta Optica Sinica*, 2016, 36(5): 0515003.
赵珊, 王彪, 唐超颖. 基于链码表示的手臂静脉特征提取与匹配[J]. 光学学报, 2016, 36(5): 0515003.
- [6] Wang K, Li Z W, Zhu C D, et al. Local stereo matching algorithm based on secondary guided filtering [J]. *Laser & Optoelectronics Progress*, 2019, 56(8): 081004.
王凯, 李志伟, 朱成德, 等. 基于二次引导滤波的局部立体匹配算法[J]. 激光与光电子学进展, 2019, 56(8): 081004.
- [7] Ouyang W L, Tombari F, Mattoccia S, et al. Performance evaluation of full search equivalent pattern matching algorithms[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(1): 127-143.
- [8] Wu P, Xu H L, Song W L. A fast NCC image matching algorithm based on waveletpyramid search strategy [J]. *Journal of Harbin Engineering University*, 2017, 38(5): 791-796.
吴鹏, 徐洪玲, 宋文龙. 结合小波金字塔的快速 NCC 图像匹配算法[J]. 哈尔滨工程大学学报, 2017, 38(5): 791-796.
- [9] Lu R Q, Ma H M. Template matching with multi-scale saliency[J]. *Optics and Precision Engineering*, 2018, 26(11): 2776-2784.
逮睿琦, 马惠敏. 多尺度显著性区域提取的模板匹配[J]. 光学精密工程, 2018, 26(11): 2776-2784.
- [10] Wu X J, Zou G H. High performance template matching algorithm based on edge geometric features [J]. *Chinese Journal of Scientific Instrument*, 2013, 34(7): 1462-1469.
吴晓军, 邹广华. 基于边缘几何特征的高性能模板匹配算法[J]. 仪器仪表学报, 2013, 34(7): 1462-1469.
- [11] Han B, Mu Z F, Le X F, et al. Fast recurrence algorithm for computing sub-image energy using normalized cross correlation[J]. *Optics and Precision Engineering*, 2018, 26(10): 2565-2574.
韩冰, 牟忠锋, 乐小峰, 等. 归一化互相关中计算基准子图能量的快速递推[J]. 光学精密工程, 2018, 26(10): 2565-2574.
- [12] Wang F B, Chu Z T, Zhu D R, et al. An improved KAZE feature detection and description algorithm [J]. *Laser & Optoelectronics Progress*, 2018, 55(9): 091007.
汪方斌, 储朱涛, 朱达荣, 等. 一种改进的 KAZE 特征检测描述算法[J]. 激光与光电子学进展, 2018, 55(9): 091007.
- [13] Dekel T, Oron S, Rubinstein M, et al. Best-buddies similarity for robust template matching [C] // 2015 IEEE Conference on Computer Vision and Pattern Recognition, June 7-12, 2015, Boston, MA, USA. New York: IEEE, 2015: 2021-2029.
- [14] Pérez P, Hue C, Vermaak J, et al. Color-based probabilistic tracking [M] // Heyden A, Sparr G, Nielsen M, et al. *Computer Vision — ECCV 2002*, Lecture Notes in Computer Science. Heidelberg, Berlin: Springer, 2002, 2350: 661-675.
- [15] Yoo J C, Han T H. Fast normalized cross-correlation [J]. *Circuits, Systems and Signal Processing*, 2009, 28(6): 819-843.
- [16] Hel-Or Y, Hel-Or H. Real-time pattern matching using projection kernels [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, 27(9): 1430-1445.
- [17] Wang G, Sun X L, Shang Y, et al. A robust template matching algorithm based on best-buddies similarity [J]. *Acta Optica Sinica*, 2017, 37(3): 0315003.
王刚, 孙晓亮, 尚洋, 等. 一种基于最佳相似点对的稳健模板匹配算法[J]. 光学学报, 2017, 37(3): 0315003.
- [18] Xia H Y, Zhao W X, Jiang F, et al. Fast template matching based on deformable best-buddies similarity measure [J]. *Multimedia Tools and Applications*, 2019, 78(9): 11905-11925.
- [19] Arbeláez P, Pont-Tuset J, Barron J, et al. Multiscale combinatorial grouping [C] // 2014 IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2014, Columbus, OH, USA. New York: IEEE, 2014: 328-335.
- [20] Wu Y, Lim J, Yang M H. Online object tracking: a benchmark [C] // 2013 IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2013, Portland, OR, USA. New York: IEEE, 2013: 2411-2418.