

基于 HSV 颜色空间的自适应窗口局部匹配算法

苏修, 陈晓冬, 徐怀远, 梁海涛, 刘依林, 汪毅, 李伟锋, 郁道银

天津大学精密仪器与光电子工程学院光电信息技术教育部重点实验室, 天津 300072

摘要 为提高立体匹配算法的效果和稳定性, 提出了一种基于色调(H)、饱和度(S)和明度(V)颜色空间的自适应聚合区域的引导滤波算法。结合图片的结构和纹理信息, 通过颜色和横向梯度的相互作用计算初始匹配代价。在 HSV 颜色空间中运用颜色和距离信息计算每一点的自适应支撑臂长, 解决了图片中红、绿、蓝 3 种颜色变化趋势相近导致无法有效反映图片信息的问题。自适应聚合区域利用中心点纵向臂上各点的横向臂进行构造, 采用引导滤波的方法在自适应聚合区域内聚合代价空间。为避免中心点邻域信息波动造成支撑窗口过小的问题, 设置了臂长的最小范围。后处理过程采用左右一致性检测结合峰比率检测的方法寻找误匹配点, 通过近邻点匹配和加权中值滤波的方法修正视差图。采用 Middlebury 平台上的标准图片进行实验, 结果表明所提算法的平均匹配误差为 5.24%, 比改进前的自适应窗口算法的匹配误差降低了 0.92%, 具有更好的边缘保持效果, 算法参数稳健性较好。

关键词 成像系统; 立体匹配; 自适应聚合区域; 引导滤波

中图分类号 TP394.41

文献标识码 A

doi: 10.3788/LOP55.031103

Adaptive Window Local Matching Algorithm Based on HSV Color Space

Su Xiu, Chen Xiaodong, Xu Huaiyuan, Liang Haitao, Liu Yilin,

Wang Yi, Li Weifeng, Yu Daoyin

Key Laboratory of the Ministry of Education on Optoelectronic Information Technology, School of Precision Instrument and Optoelectronics Engineering, Tianjin University, Tianjin 300072, China

Abstract In order to improve the effect and stability of stereo matching algorithm, we propose a guided filtering algorithm based on adaptive aggregation region in Hue-Saturation-Value (HSV) color space. We calculate initial matching cost by the interaction of color and transverse gradient in combination with the structure and texture information of the image. Then, we calculate the length of the adaptive support arm of each point based on the color and distance information in HSV color space, which solves the problem that the change trend of the red, green and blue colors in the picture is similar and cannot effectively reflect the picture information. We construct the adaptive aggregation region using a transverse arm at each point on the longitudinal arm of the center point, and aggregate the cost space in the adaptive aggregation region by guided filtering. In order to avoid the problem that the support window is too small due to the fluctuation of the neighborhood information, we set the minimum range of the arm length. In the process of post-processing, we use the left and right consistency detection and the peak ratio detection method to find mismatching points, and correct the disparity map by the nearest neighbor matching and weighted median filtering. The experiments are carried out on standard images of Middlebury platform. Results show that average matching error of the proposed algorithm is 5.24%, and the matching error is reduced by 0.92% compared with the that of pre-improvement adaptive window algorithm. The proposed algorithm has better edge preservation effect and the robustness of the algorithm parameters is good.

Key words imaging systems; stereo matching; adaptive aggregation region; guided filtering

OCIS codes 110.3010; 120.2440; 150.1135

收稿日期: 2017-09-13; 收到修改稿日期: 2017-10-11

作者简介: 苏修(1994—), 男, 硕士研究生, 主要从事三维图像处理方面的研究。E-mail: suxiu@tju.edu.cn

导师简介: 陈晓冬(1975—), 男, 博士, 教授, 主要从事光电成像技术及系统方面的研究。

E-mail: xdchen@tju.edu.cn(通信联系人)

1 引言

立体匹配是近年来计算机视觉领域的研究热点之一,其核心思想是在左右图像中寻找对应点,求取视差图以获得空间深度信息。目前,立体匹配算法主要分为全局立体匹配和局部立体匹配两类^[1]。全局立体匹配通过数据项和平滑项能量函数将匹配问题转化为能量最小化问题。常用的全局立体匹配算法有动态规划法^[2]、图割法^[3]、马尔可夫随机场^[4]、置信传播等^[5]。这类方法的计算复杂度高,实时性较差。局部立体匹配算法是一种基于窗口的滤波算法,其运行时间短,发展潜力大。

在代价计算过程中,传统的局部立体匹配方法主要包括基于灰度绝对差(AD)、平方差(SD)、梯度差^[6]、对比度^[7]和归一化互相关(NCC)^[8]的方法,以及非参数变换^[9](如 Rank, census)等方法。在代价聚合过程中,很多方法如可变支撑窗口^[10-11]、自适应支撑权重^[12-13]、引导滤波等方法均被提出^[14],其中 Yoon 等^[12]于 2006 年提出了自适应权重的局部立体匹配算法,该算法在匹配效果上取得了接近全局匹配的效果。但是利用这类方法进行计算时,仅考虑颜色和距离的作用,不能较好地保持物体边缘信息计算的正确性,无法实现并行计算,复杂度较高。2013 年,He 等^[14]提出了引导滤波算法,取得了类似自适应立体匹配算法的效果,通过引入并行运算,

降低了算法的复杂度;该方法中的滤波窗口大小是固定的,然而实际情况下图片的大小、颜色、空间距离等都会对窗口的选择产生一定的影响,因此采用固定窗口的方法不可避免地会引入误差。Xu 等^[15]提出了自适应形状的支撑窗口方法,该方法在一定程度上解决了固定窗口的问题,但是仅通过图片的红、绿、蓝(RGB)颜色权重进行对比,而图片的 RGB 颜色变化趋势具有相似性,因此该方法在构造与中心点相关的滤波区域时存在类似信息重复计算的问题,容易引入无关点的干扰,并且该方法也没有考虑聚合窗口中点与点之间距离的影响。

本文提出了一种基于色调(H)、饱和度(S)和明度(V)颜色空间的局部立体匹配算法。该算法从图片的亮度、色度、饱和度 3 个不同维度为每个中心像素选出滤波区域,通过代价聚合过程获得更高的匹配精度。由于距离中心点近的像素点更可能与中心点具有相同视差,因此将距离中心点较近的区域加入自适应支撑窗口中,从而有效提高立体匹配精度。

2 算法描述

所提方法分 4 个步骤:匹配代价计算,聚合区域选取,代价聚集以及视差选取和后处理过程,算法框架如图 1 所示。ARSW 代表自适应聚合区域, WTA 为 Winner-Take-All。

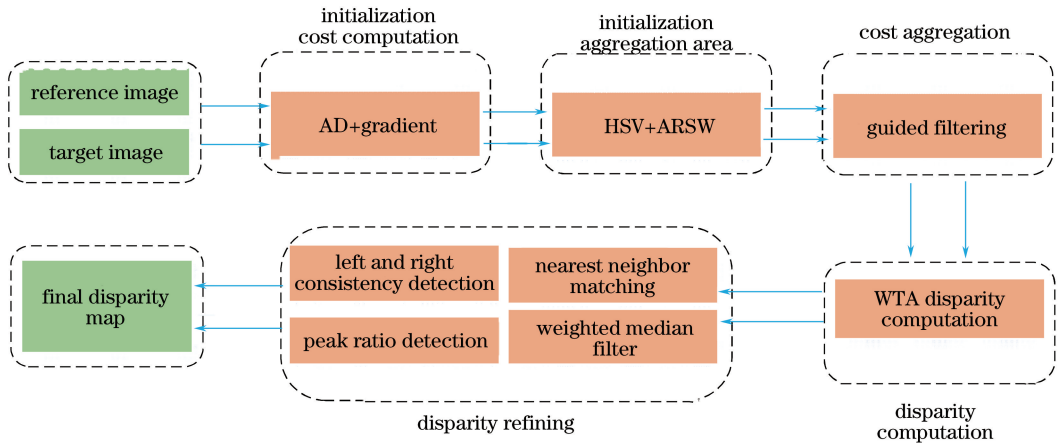


图 1 所提算法框图

Fig. 1 Block diagram of proposed algorithm

2.1 匹配代价计算

匹配代价主要考查左右图像中对应点之间的相似性。多数立体匹配算法会将灰度差绝对值和、横向梯度差绝对值和、统计变换(CT)的汉明距离等结合使用^[16]。在求取视差图时综合了色度绝对差和横向梯度差。色度绝对差表征匹配的概率,利用色

度绝对差能提高规律纹理区域的匹配精度。图像的梯度值包含丰富的结构信息,梯度差的应用能区分待匹配点附近的结构差异。左右图的对应匹配代价值可表示为

$$C(p, d) = (1 - \alpha) \min(C_{AD}, T_{AD}) + \alpha \min(C_{GRAD}, T_{GRAD}), \quad (1)$$

式中 α 为调节色度绝对值差和横向梯度绝对值差权重的比例系数, T_{AD} 和 T_{GRAD} 分别为色度绝对值差和横向梯度绝对值差的截断值。令 c 为 RGB 颜色空间中 R、G、B 颜色分量中的一种, p 为参考图中的一点, d 为目标图中对应点 p' 与点 p 在 x 轴方向的距离, $C_{AD}(p, d)$ 和 $C_{GRAD}(p, d)$ 分别为色度绝对值差和横向梯度绝对值差, 可表示为

$$C_{AD}(p, d) = \sum_{c \in \{R, G, B\}} |I_c(p) - \hat{I}_c(p')|, \quad (2)$$

$$C_{GRAD}(p, d) = |\nabla_x I'(p) - \nabla_x \hat{I}'(p')|, \quad (3)$$

式中 $I_c(p)$ 、 $\hat{I}_c(p')$ 分别表示左右视图中点 p 和 p' 的颜色强度, $I'(p)$ 、 $\hat{I}'(p')$ 分别表示左右视图所对

应的灰度图中点 p 和 p' 的强度。假设参考图中点 p 的位置为 (x_p, y_p) , 其在目标图中的对应点 p' 的位置为 $(x_p - d, y_p)$ 。截断值能有效抑制噪声与正常值之差的极大值, 有利于后续的滤波过程。

2.2 聚合区域的选取

2.2.1 HSV 颜色空间

目前立体匹配算法中聚合区域的选取大多基于 RGB 颜色空间, 但是由于在同一场景中 R、G、B 颜色分量的变化具有一定的相似性, 因此该方法并不能有效提取图片的信息。图 2 中展示了对于同一图片的 R、G、B 颜色分量和 H、S、V 颜色分量的灰度图对比。

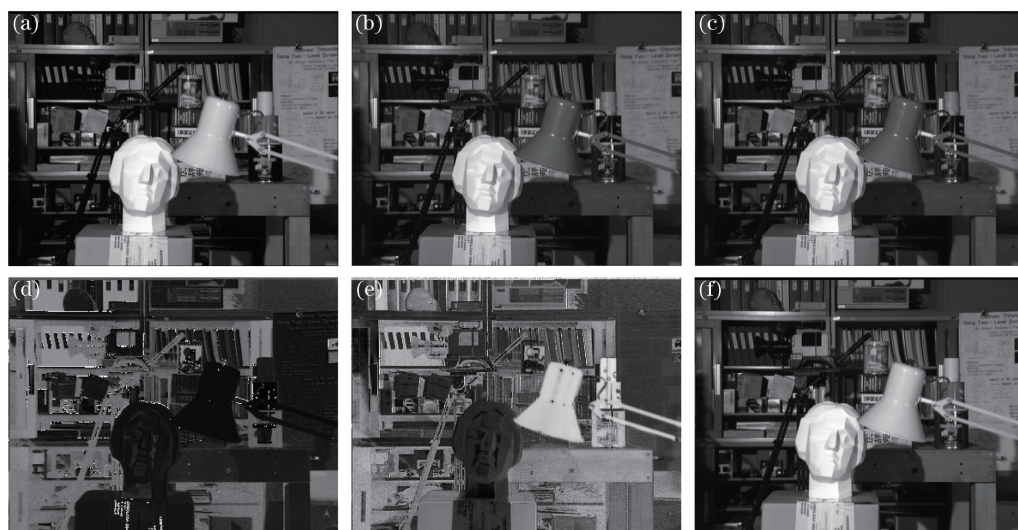


图 2 Tsukuba 颜色分量灰度图。(a) R 分量; (b) G 分量; (c) B 分量; (d) H 分量; (e) S 分量; (f) V 分量

Fig. 2 Color component grayscale of Tsukuba. (a) R component; (b) G component; (c) B component; (d) H component; (e) S component; (f) V component

从图 2 中可以看出, 图片的 R、G、B 分量在颜色变化上具有很好的一致性, 因此在 RGB 颜色空间中求取聚合区域时, 信息的重复计算导致无法获取合适的聚合区域。HSV 颜色空间从色调、饱和度和明度 3 个方面提取图像信息, 提高了图像的分辨水平, 因此在 HSV 颜色空间中能有效提取出图片的信息。H 分量突出反映了图像的色彩变化情况, 能有效区分图片的前景和背景; S 分量突出体现了图像边界和颜色变化; V 分量则反映了图像中的亮度变化。

2.2.2 自适应聚合区域

首先将图片从 RGB 颜色空间转换到 HSV 颜色空间, 采用 3×3 的中值滤波函数平滑处理 H 分量图和 S 分量图, 以排除噪声对窗口构造的影响。为了找出深度相近的均匀区域, 进行如下假设: 图片中亮度、色度和饱和度之一变化较大的区域发生了

空间深度或图片结构的明显变化, 因此在同一滤波窗口内的像素具有类似的视差和纹理结构。

通过确定每个点 p 所在位置 (x_p, y_p) 在 4 个方向的臂长 $\{l_p^r, l_p^l, l_p^u, l_p^d\}$ ($l_p^r, l_p^l, l_p^u, l_p^d$ 分别代表右、上、左、下臂长) 来构造聚合区域, 臂长的大小表示像素点的个数。设 $H(p)$ 表示每个点的横向臂, $V(p)$ 表示以点 p 为中心的竖向臂。那么, 以点 p 为中心的区域为

$$N_p = \bigcup_{q \in V(p)} H(q), \quad (4)$$

式中 q 为以点 p 为中心的位于臂上的点。

接着, 根据亮度、色度和饱和度 3 个分量的变化情况来决定每个点 p 的臂长。由于距离中心点越近的对中心点的影响越大, 在构造臂长时, 为了防止聚合区域过小带来误差(如边界区域附近的点), 设臂

长最小值为 L_{\min} , 同时设臂长的最大值为 L_{\max} 。考虑到 H、S、V 分量分别代表同一图片的不同维度, 因此它们对图片信息的反映权重是不同的, 设 H、S、V 分量的权重分别为 h_0 、 s_0 和 v_0 , 即 α_c 的分量。以右臂长 l_p^r 的计算为例, 设最佳臂长为 l^* , 当 $l \in [L_{\min}, l^*]$ 时, $\max_{c \in \{H, S, V\}} (\alpha_c |I_c(p) - I_c(p_l)|) \leq \tau$, 其中 $\alpha_c \in \{h_0, s_0, v_0\}$; 当 $l = l^* + 1$ 时, $\max_{c \in \{H, S, V\}} (\alpha_c |I_c(p) - I_c(p_l)|) > \tau$, 若 $l^* < L_{\min}$, 设 $l^* = L_{\min}$ 。因此右臂上的每一点用 $\delta(p, p_l)$ 表示, $\delta(p, p_l) = 1$ 表示该点 p_l 位于以点 p 为中心建立的臂长上, $\delta(p, p_l) = 0$ 表示其不存在于点 p 为中心的臂长上, 即

$$\begin{cases} \delta(p, p_l) = 1, & l \in (0, l^*] \\ \delta(p, p_l) = 0, & l \in (l^*, +\infty) \end{cases} \quad (5)$$

l_p^l, l_p^r, l_p^d 同样可以用上面提到的方法求出, 结

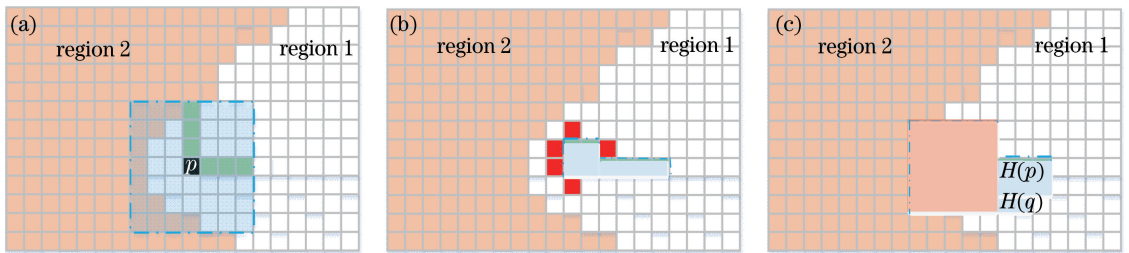


图 3 不同的聚合区域对比。(a)固定区域;(b)自适应聚合区域;(c)所提方法选取的聚合区域

Fig. 3 Comparison of different aggregation regions. (a) Fixed region; (b) adaptive aggregation region; (c) aggregation region selected by proposed method

2.3 代价聚合

代价聚合过程是利用像素周围结构的信息对中心像素进行滤波, 通过引入像素间的关联性, 提高视差的准确度。采用引导滤波算法, 针对代价空间中的一点 i , 引入 W_{ij} 权重因子, j 代表代价空间中的另一点, 结合周围同质区域的点优化中心点的权值, 得到

$$q(i) = \sum_j W_{ij}^{GF} p(j), \quad (6)$$

$$W_{ij}^{GF} =$$

$$\frac{1}{|\omega|} \sum_{k: (i,j) \in \omega_k} \left\{ 1 + \frac{[I(j) - \mu_k][I(i) - \mu_k]}{\sigma_k^2 + \epsilon} \right\}, \quad (7)$$

式中 W_{ij}^{GF} 表示 i 点与 j 点间的滤波权重, $I(i)$ 和 $I(j)$ 分别为 i 点和 j 点处的灰度值, μ_k 和 σ_k^2 分别为 I 在窗口 ω_k 内的均值和方差, 窗口内的像素点数为 $|\omega|$, ϵ 为平滑因子, $p(j)$ 为初始匹配代价值, $q(i)$ 为滤波后输出的代价值。中心点在 k 点, 窗口 ω_k 采用第 2.2 节中提出的方法获得。

合(4)式可构建所有点 p 的自适应聚合窗口 N_p 。

图 3 对比了不同的聚合区域构造方法。图 3(a)展示了矩形区域的聚合方式^[14], 该方法会引入一定量的无关点, 导致代价聚合产生错误的结果。图 3(b)展示了自适应聚合区域的聚合方式^[15], 其中红点是边界附近图像信息变化剧烈的点, 这些特殊点的干扰导致聚合区域非常小, 无法正确获得滤波的输出结果。图 3(c)展示了所提方法选取的聚合区域, 其中红色方框内包含的点表示距离 p 点较近的信任区域, 绿色方框内包含的点是基于 HSV 颜色空间的自适应聚合窗口的方式构造而成, 从图中可以看出, 所提方法在保证聚合区域具有足够尺寸的同时, 能尽可能多的包含同质区域的像素, 从而获得正确的滤波结果。

2.4 视差处理和后处理

视差选取整合视差空间的信息, 采用 WTA 的方法求取视差图。选取的最优值为

$$d_p = \arg \min_{d \in R} C_{\text{agg}}(p, d), \quad (8)$$

式中 R 为视差的取值范围, $C_{\text{agg}}(p, d)$ 为初始视差值。

初始视差图中仍存在一些遮挡点和误匹配点, 可通过优化视差图的方法去除。首先通过左右一致性检测找出图中视差变化明显区域的误匹配点, 然后在视差变化不明显区域, 利用峰比率检测搜寻不稳定点, 即

$$M_p^{\text{PKR}} = \frac{|C_{p,1} - C_{p,2}|}{C_{p,2}}, \quad (9)$$

式中 M_p^{PKR} 为点 p 的峰比率, $C_{p,1}$ 为代价空间中最小的匹配代价值, $C_{p,2}$ 为次小的匹配代价值。次小值越接近最小值, 该点的峰比率越接近于零, 则该最小值越不可信。峰比率低于特定阈值 η_{PKR} 的点称为不稳定点。

采用近邻点法寻找误匹配点对应的正确匹配

点,即以中心点为核心,进行横向和竖向正确匹配点搜索,即

$$d_p^* = \begin{cases} d_p^*, & \text{if } d_{lr} \text{ and } d_{ud} \text{ not exist} \\ d_{lr}, & \text{else if only } d_{lr} \text{ exist} \\ d_{ud}, & \text{else if only } d_{ud} \text{ exist} \\ \min(d_{lr}, d_{ud}), & \text{otherwise} \end{cases}, \quad (10)$$

式中 d_{lr} 代表 $\min(d_l, d_r)$, 其中 d_l 和 d_r 为横向最近的正确点的视差; d_{ud} 表示 $\min(d_u, d_d)$, 其中 d_u 和 d_d 为竖向最近的正确点的视差。最后采用加权中值滤波的方法对修正后的误匹配点和检测出的不稳定点进行进一步优化^[17]。

3 仿真结果与分析

3.1 基于理想图像的仿真

算法实现的硬件平台中央处理器(CPU)为 Intel(R) Core(TM) i5-7400, 显卡(GPU)型号为 GeForce GTX 750 Ti。在 Ubuntu Linux 系统中编写代码,运用统一计算设备架构(CUDA)进行内存优化和并行运算处理,采用 Nsight Eclipse Edition

软件进行调试。仿真中,采用 Middlebury 立体匹配算法测试平台提供的标准立体图像,以验证算法的有效性。网站提出的 4 组基准彩色图像对 Tsukuba、Venus、Teddy 和 Cones 的分辨率分别为 $384 \text{ pixel} \times 288 \text{ pixel}$ 、 $434 \text{ pixel} \times 383 \text{ pixel}$ 、 $450 \text{ pixel} \times 375 \text{ pixel}$ 和 $450 \text{ pixel} \times 375 \text{ pixel}$, 视差范围分别为 $0 \sim 15 \text{ pixel}$ 、 $0 \sim 32 \text{ pixel}$ 、 $0 \sim 59 \text{ pixel}$ 和 $0 \sim 59 \text{ pixel}$ 。其他参数设置如下: $\alpha = 0.89$, $T_{AD} = 0.0275$, $T_{GRAD} = 0.0078$, $h_0 = 0.85$, $s_0 = 0.84$, $v_0 = 1.4$, $\tau = 0.1$, $\eta_{PKR} = 0.0219$, $L_{\min} = 4$, $L_{\max} = 16$, $r = 31$ 。除了视差范围不同外,其余参数在测试过程中均保持不变。

图 4 为该算法得到的视差效果图, 图像的顺序由上至下依次是 Tsukuba、Venus、Teddy 和 Cones。每行图的第二列为真实视差图, 第 3 列为该算法的视差图, 第 4 列为自适应立体匹配方法的视差图^[7], 第 5 列为 ASSW 方法的视差图^[15]。通过比较仿真结果可以看出, 相比于其他类似的局部立体匹配算法, 该算法具有较高的匹配精度和更好的保边效果, 如图中红色方框内的图像边缘所示。

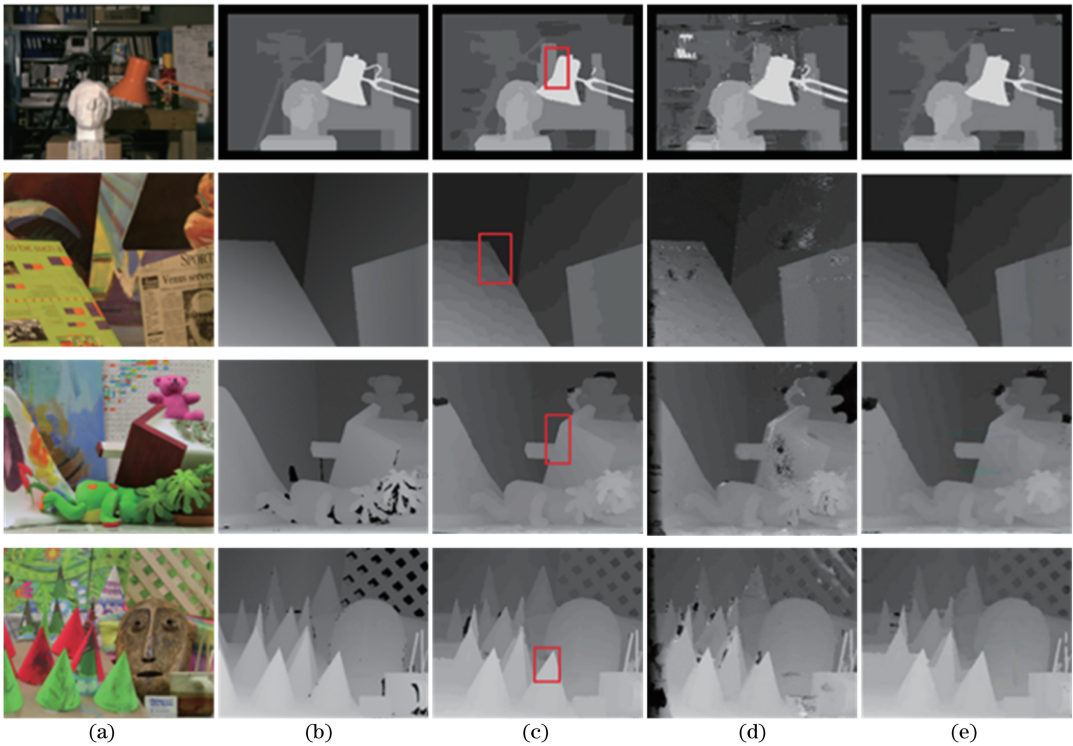


图 4 (a) 参考图像; (b) 真实视差图; (c) 所提算法, (d) 自适应立体匹配方法和 (e) ASSW 方法对 Middlebury 图像对的仿真结果
Fig. 4 (a) Reference images; (b) actual disparity maps; simulation results obtained by (c) proposed algorithm, (d) adaptive stereo matching method and (e) ASSW method for Middlebury stereo pairs

当阈值为 1 时, 所提算法和类似算法的误匹配率比较结果如表 1 所示。Middlebury 平台上提供

的检验算法有效性的标准模板有 3 种, 分别为非遮挡区域误匹配像素比 (Noocc)、总误匹配像素比

(All)和深度不连续区域误匹配像素比(Disc)。通过对比可知,所提算法能获得较高的匹配精度,平均匹配误差低于很多局部算法(如 Adapt Weight 和 Variable Cross),以及半全局算法(Semi Glob)和全局算法[如 GC + occ 和 SMPF (stereo matching based on particle filters)]所提算法的整体匹配正确

率明显高于其他几种算法。需要特别指出的是,部分算法对 Tsukuba 图的匹配效果优于所提算法,这是因为 Tsukuba 图的深度不连续性程度高,图片中前景和背景变化快,相对于所提的局部立体匹配算法,全局立体匹配算法的计算复杂度高,更能处理复杂变化的区域,从而具有更好的匹配效果。

表 1 阈值为 1 时的误匹配像素平均百分比

Table 1 Average percentage of mismatched pixels with threshold of 1

%

Algorithm	Tsukuba			Venus			Teddy			Cones			Average error
	Noocc	All	Disc	Noocc	All	Disc	Noocc	All	Disc	Noocc	All	Disc	
Proposed	1.58	1.86	5.83	0.27	0.46	2.40	6.10	11.30	14.50	2.71	8.42	7.50	5.24
ASSW ^[13]	1.81	2.17	7.85	0.32	0.51	3.73	7.02	12.50	17.40	3.21	8.40	8.99	6.16
Adapt weight ^[12]	1.38	1.85	6.90	0.71	1.19	6.13	7.88	13.30	18.60	3.97	9.79	8.26	6.67
Cost filter ^[18]	1.51	1.85	7.61	0.20	0.39	2.42	6.16	11.80	16.00	2.71	8.24	7.66	5.55
Variable cross ^[19]	1.99	2.65	6.77	0.62	0.96	3.20	9.75	15.10	18.20	6.28	12.70	12.90	7.60
SMPF ^[20]	0.98	1.53	5.31	0.25	0.69	2.60	9.93	14.50	22.60	6.51	13.10	14.80	7.73
GC+occ ^[21]	1.19	2.01	6.24	1.64	2.19	6.75	11.20	17.40	19.80	5.36	12.40	13.00	8.26
VSW (virtual support window) ^[22]	1.62	1.88	6.98	0.47	0.81	3.40	8.67	13.30	18.00	3.37	8.82	8.12	6.29

表 2 中展示了在其他参数保持不变的情况下,误差阈值为 0.5 时的效果对比。可见,所提算法依然具有最低的平均匹配误差。同时,因为 Venus 的

图片深度结构最平滑,该算法在 Noocc、All、Disc 方面均具有最好的效果。总体来说,该方法在立体匹配方面具有更高的匹配正确率和更好的保边效果。

表 2 阈值为 0.5 时的误匹配像素平均百分比

Table 2 Average percentage of mismatched pixels with threshold of 0.5

%

Algorithm	Tsukuba			Venus			Teddy			Cones			Average error
	Noocc	All	Disc	Noocc	All	Disc	Noocc	All	Disc	Noocc	All	Disc	
Proposed	13.70	14.00	16.00	7.08	7.49	11.50	12.30	18.50	24.00	8.91	14.90	15.10	13.60
ASSW ^[13]	11.90	12.50	17.00	6.66	7.11	12.30	12.70	19.10	26.60	9.01	14.50	16.30	13.80
Adapt weight ^[12]	18.10	18.80	18.60	7.77	8.40	15.80	17.60	23.90	34.00	14.00	19.70	20.60	18.10
Variable cross ^[19]	24.50	25.10	21.50	9.03	9.59	13.80	18.80	25.10	31.40	16.10	22.10	22.40	19.90
SMPF ^[20]	19.50	20.10	16.60	8.90	9.56	15.10	20.50	25.90	38.20	13.90	20.50	26.20	19.60
GC+occ ^[21]	6.10	7.11	14.60	10.70	11.30	16.90	23.70	30.10	34.60	12.20	19.20	21.90	17.40
VSW ^[22]	19.20	19.50	18.50	8.17	8.65	13.20	17.40	23.20	31.40	13.10	18.30	20.40	17.60

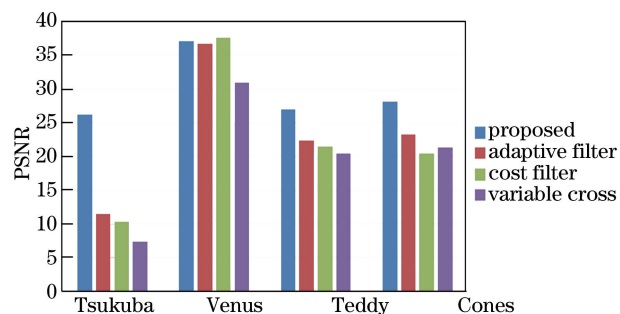


图 5 所提算法与其他类似方法的 PSNR 对比
Fig. 5 PSNR comparison of proposed algorithm with other similar methods

为了证明该方法的优越性,除比较匹配误差率之外,将所提方法与其他类似方法的 PSNR 进行对比。PSNR 越大,证明算法所得视差图越接近于真实视差图。图 5 展示了所提算法与 adaptive filter^[23]、cost filter^[18] 和 variable cross^[19] 算法的 PSNR 数据对比。可以看出,所提方法的 PSNR 数据普遍优于其他类似算法。

图 6 给出了所提方法与类似方法的运行时间对比。所提方法的运算速度慢于 variable cross 方法,但优于其他类似的立体匹配算法。所提方法的运算速度较快,能够满足实际运用的要求。

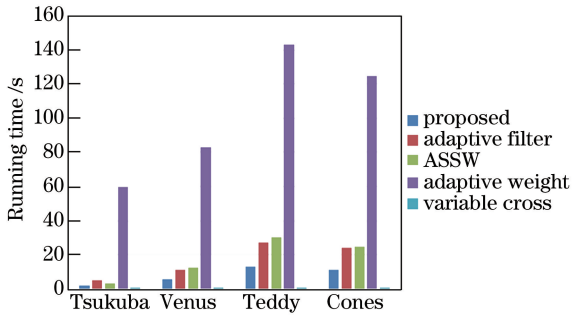


图6 所提算法与其他类似方法的运行时间对比
Fig. 6 Running time comparison of proposed algorithm and other similar methods

3.2 后处理参数的稳健性仿真

图7中展示了所提算法参数的稳健性仿真结果。图7(a)展示了 $\tau=0.1$ 时, L_{max} 的值从9变化到20的结果,由图可知所提算法对支撑窗口的最大臂

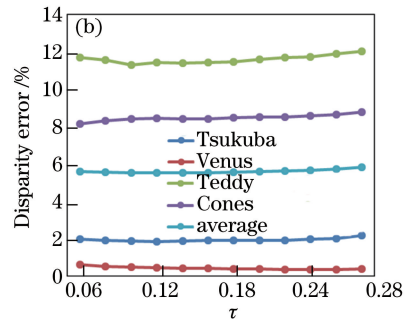
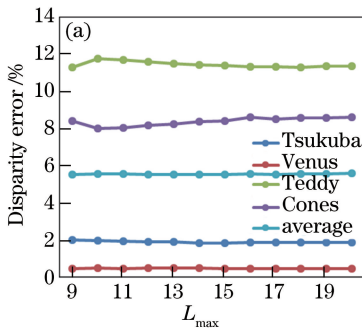


图7 参数(a) L_{max} 和(b) τ 的稳健性测试

Fig. 7 Robustness tests for parameters (a) L_{max} and (b) τ

4 结 论

提出了一种基于 HSV 颜色空间的自适应聚合区域的局部匹配算法,用于解决目前普遍出现的 RGB 颜色空间无法有效反映图片信息的问题。所提算法基于 HSV 颜色空间计算自适应聚合区域,设置了支撑臂长的最小范围,有效避免了图片信息剧烈变化的区域,提高了立体匹配的准确度。仿真结果表明,该算法的匹配精度优于其他类似的局部立体匹配方法,对物体边缘的处理效果较好,参数的稳健性较好。

参 考 文 献

[1] Scharstein D, Szeliski R, Zabih R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms [C]. Stereo and Multi-Baseline Vision, 2001: 7184510.
[2] Chang X F, Zhou Z, Wang L, *et al.* Real-time accurate stereo matching using modified two-pass

长 L_{max} 的选取并不敏感,4 幅图片的匹配误差几乎不随最大臂长的大小而改变。当固定 $L_{max} = 16$ 时, τ 的值从 0.06 变化到 0.28 的结果如图 7(b)所示,误差率几乎保持不变。

改变 L_{max} 或 τ 的数值都不会对 4 幅图的误差率产生显著影响,证明了所提算法的稳健性。由于 Teddy 和 Cones 图片具有复杂结构,并且视差范围较广,在构建基于 HSV 颜色空间的自适应聚合区域时,易受到快速变化的视差的影响。相反, Venus 和 Tsukuba 图片的深度变化范围较小,因此构建自适应聚合区域时产生的错误更少,平均误匹配率也会更低。从图中可以看出,误差率从低到高分别是 Venus, Tsukuba, Cones 和 Teddy,参数变化时误差变化不大,说明所提方法具有很好的稳健性。

aggregation and winner-take-all guided dynamic programming [C]. International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, 2011: 12122106.

[3] Papadakis N, Caselles V. Multi-label depth estimation for graph cuts stereo problems [J]. Journal of Mathematical Imaging & Vision, 2010, 38(1): 70-82.
[4] Li J, Qian W X, Chen Q, *et al.* An efficient stereo matching method based on Bayesian theory [J]. Laser & Optoelectronics Progress, 2014, 51(10): 101001. 李娇, 钱惟贤, 陈钱, 等. 一种基于贝叶斯理论的高效立体匹配方法 [J]. 激光与光电子学进展, 2014, 51(10): 101001.
[5] Besse F, Rother C, Fitzgibbon A, *et al.* PMBP: PatchMatch belief propagation for correspondence field estimation [J]. International Journal of Computer Vision, 2014, 110(1): 2-13.
[6] Zhu S P, Li Z. A stereo matching algorithm using improved gradient and adaptive window [J]. Acta Optica Sinica, 2015, 35(1): 0110003.

- 祝世平, 李政. 基于改进梯度和自适应窗口的立体匹配算法[J]. 光学学报, 2015, 35(1): 0110003.
- [7] Liu T L, Luo L M. High efficiency feature-based and robust local stereo matching algorithm in medical optics[J]. Laser & Optoelectronics Progress, 2010, 47(5): 051501.
刘天亮, 罗立民. 医用光学中基于局部特征的高效稳健立体匹配算法[J]. 激光与光电子学进展, 2010, 47(5): 051501.
- [8] Hirschmuller H. Accurate and efficient stereo processing by semi-global matching and mutual information[C]. IEEE Computer Society Conference on Computer Vision & Pattern Recognition, 2005: 8624110.
- [9] Zhu S P, Yan L N, Li Z. Stereo matching algorithm based on improved census transform and dynamic programming[J]. Acta Optica Sinica, 2016, 36(4): 0415001.
祝世平, 闫利那, 李政. 基于改进 Census 变换和动态规划的立体匹配算法[J]. 光学学报, 2016, 36(4): 0415001.
- [10] Hu W D, Zhang K, Sun L F, *et al.* Virtual support window for adaptive-weight stereo matching [C]. Visual Communications and Image Processing, 2011: 12493695.
- [11] Veksler O. Fast variable window for stereo correspondence using integral images [C]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003: 7769583.
- [12] Yoon K J, Kweon I S. Adaptive support-weight approach for correspondence search [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2006, 28(4): 650-656.
- [13] Gong W B, Gu G H, Qian W X, *et al.* Stereo matching algorithm based on the inter color correlation and adaptive support weight[J]. Chinese Journal of Lasers, 2014, 41(8): 0812001.
龚文彪, 顾国华, 钱惟贤, 等. 基于颜色内相关和自适应支撑权重的立体匹配算法[J]. 中国激光, 2014, 41(8): 0812001.
- [14] He K, Sun J, Tang X. Guided image filtering[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2013, 35(6): 1397-1409.
- [15] Xu Y F, Zhao Y, Ji M Q. Local stereo matching with adaptive shape support window based cost aggregation[J]. Applied Optics, 2014, 53(29): 6885-6892.
- [16] Chang T A, Lu X, Yang J F. Robust stereo matching with trinary cross color census and triple image-based refinements[J]. Eurasip Journal on Advances in Signal Processing, 2017: 27.
- [17] Ma Z Y, He K M, Wei Y C, *et al.* Constant time weighted median filtering for stereo matching and beyond[C]. IEEE International Conference on Computer Vision, 2014: 14144919.
- [18] Hosni A, Rhemann C, Bleyer M, *et al.* Fast cost-volume filtering for visual correspondence and beyond [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(2): 504-511.
- [19] Zhang K, Lu J B, Lafruit G. Cross-based local stereo matching using orthogonal integral images[J]. IEEE Transactions on Circuits & Systems for Video Technology, 2009, 19(7): 1073-1079.
- [20] Ploumpis S, Amanatiadis A, Gasteratos A. A stereo matching approach based on particle filters and scattered control landmarks [J]. Image and Vision Computing, 2015, 38: 13-23.
- [21] Kolmogorov V, Zabih R. Computing visual correspondence with occlusions using graph cuts[C]. Eighth IEEE International Conference on Computer Vision, 2001: 7024320.
- [22] Hu W D, Zhang K, Sun L F, *et al.* Virtual support window for adaptive-weight stereo matching [C]. Visual Communications and Image Processing, 2011: 12493695.
- [23] Zhu S P, Yan L N. Local stereo matching algorithm with efficient matching cost and adaptive guided image filter[J]. The Visual Computer, 2017, 33(9): 1087-1122.