

DOI:10.16136/j.joel.2019.09.0085

# 融合梯度特性与置信度的立体匹配算法

范海瑞, 杨帆\*, 潘旭冉, 李靖, 潘国峰

(河北工业大学 电子信息工程学院, 天津 300401)

针对什么问题(两个问题), 提出什么算法(论文题目)。

摘要: 针对现有局部立体匹配算法在计算匹配代价时, 不能很好区分强弱纹理区域, 及在视差计算过程中, 不能很好的解决视差歧义问题, 提出一种融合梯度特性与置信度的立体匹配算法。首先计算梯度特征, 并根据梯度特征信息选择匹配代价计算的匹配窗口, 针对强弱不同纹理区域选择不同尺寸的匹配窗口, 有效的提高了立体匹配精度, 降低了误匹配率; 然后在视差计算中引入置信度约束条件, 解决了视差计算中视差歧义的问题, 提高了立体匹配算法的稳定性与精度; 最后使用水平与垂直方向交叉区域检测进行奇异值的修正。实验结果表明, 该算法在 Middlebury 数据集中 31 对立体图像的平均误匹配率为 7.96%, 有效的提高了立体匹配精度。

从操作层面描述该算法和效果

关键词: 机器视觉; 立体匹配; 梯度变换; 置信度; 引导滤波

中图分类号: TP391.4 文献标识码: A 文章编号: 1005-0086(2019)09-0985-09

拿实验结果表现有效性

## Stereo matching algorithms fusing gradient characteristic and confidence

FAN Hai-rui, YANG Fan\*, PAN Xu-ran, LI Jing, PAN Guo-feng

(School of Electronic and Information Engineering, Hebei University of Technology, Tianjin 300401, China)

**Abstract:** As the existing local stereo matching algorithms can not distinguish the strong and weak different texture regions well, and can not solve the disparity ambiguity problem in the step of disparity selection, a stereo matching algorithm based on gradient characteristics and confidence is proposed in this paper. Firstly, the gradient features are calculated, and the support window which is used to calculating the matching cost is selected according to gradient feature information, and different sizes of support windows can be selected according to strong or weak different texture, which effectively improves the accuracy of stereo matching and effectively reduces the false matching rate when in the process of calculating matching cost. Secondly, the confidence constraint is introduced into disparity calculation to solve disparity ambiguity in the step of disparity calculation and improve the stability of stereo matching algorithm. Finally, the singular value which is detected by left and right consistency detection is corrected by using cross-region detection in horizontal and vertical directions. The experimental results show that the average mismatch rate of 31 pairs of stereo images in the Middlebury dataset is 7.96%. Compared with common stereo match algorithms, the result show proposed algorithm is effectively improves the stereo matching accuracy.

**Key words:** machine vision; stereo matching; gradient transform; confidence; guided filtering

## 1 引言

立体视觉是计算机视觉中的重要方向, 被广泛应用于无人驾驶、增强现实、虚拟现实和多场景三维重建等热门领域中<sup>[1]</sup>; 立体匹配是立体视觉

中非常重要的一步, 它通过寻找同一场景在不同视角下的两幅或者多幅图像的匹配像素点, 然后通过匹配像素恢复深度信息, 从而可以建立三维稠密地图<sup>[2]</sup>。Scharstei 等<sup>[3]</sup>人建立立体匹配算法理论框架, 将算法分为四个步骤: 匹配代价计算,

\* E-mail: commander jy@163.com

收稿日期: 2019-04-03 修订日期: 2019-06-22

基金项目: 国家中长期科技发展规划 02 科技重大专项资助项目(2016ZX02301003-004-007)、天津市自然科学基金项目(17JCTPJC54500)和河北省高等学校科学技术研究重点项目(ZD2016123)资助项目

匹配代价聚合,视差计算和视差后处理。根据这个理论框架,当前算法可以分为三类:全局立体匹配算法、局部立体匹配算法和基于监督学习的立体匹配算法。

常见的全局算法有置信传播(BP)<sup>[4]</sup>、图割法(GC)<sup>[5]</sup>、最小生成树(MST)<sup>[6]</sup>、MST-DC<sup>[7]</sup>和分割树(ST)<sup>[8]</sup>等,全局算法通过优化全局能量函数获得最终视差,精度较高但计算复杂,有时会出现局部最优。基于监督学习的算法主要通过机器学习或深度学习进行,常见的有 MC-CNN 算法<sup>[9]</sup>及改进 MC-CNN<sup>[9]</sup>、DispNet 算法<sup>[11]</sup>和 DRL 算法<sup>[12]</sup>等,这些算法都较为复杂,同时不能很好的解决遮挡问题,在某些特定场合受限制。常见的局部算法有 NCC、自适应窗口(AW)<sup>[13]</sup>和 PatchStereo<sup>[14]</sup>等,局部算法使用局部信息求取匹配并计算视差,精度略低于全局算法,但具有复杂度低、效率高和便于实现和便于在硬件上实现等优点,较为常用。

综合各类算法的优缺点,提出一种融合梯度特性与置信度的立体匹配算法。

## 2 本文立体匹配算法总体框架

本文提出的融合梯度特性与置信度的立体匹配算法流程如图1所示。算法通过基于梯度特性的自

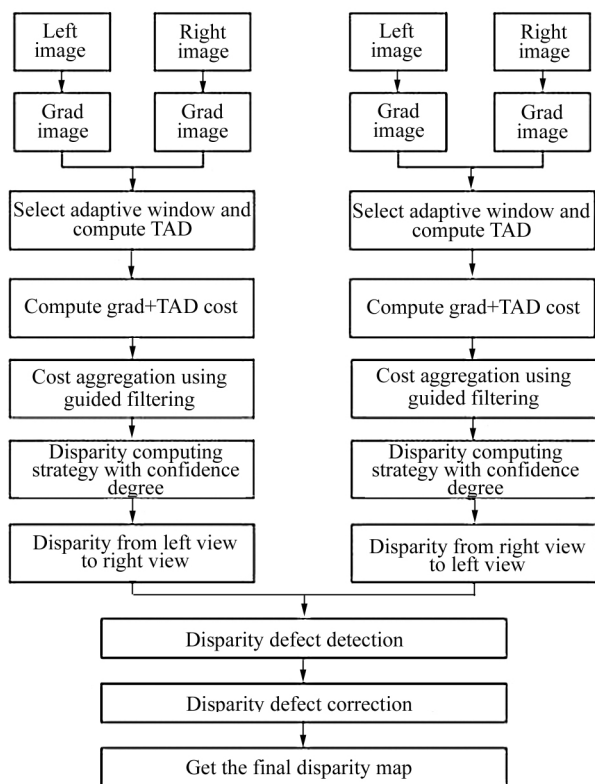


图1 融合梯度特性与置信度的立体匹配算法框架图

Fig. 1 Framework of Stereo Matching Algorithms Fusing Gradient Characteristics and Confidence

适应窗口选择策略计算匹配代价,使用引导滤波算法进行匹配代价聚合,根据基于置信度的视差计算策略获得初始视差图,最后使用本文提出的视差优化方法进行视差后处理,获得高质量的视差图。

## 3 算法描述

本文算法主要包括匹配代价计算、匹配代价聚合、视差计算和视差优化四个步骤。

### 3.1 匹配代价计算

匹配代价是指左右视图对需要进行匹配的像素点之间相似性的度量。通常用基于像素灰度值的 SSD(Sum of Squared Differences, 误差平方和)算法计算匹配代价,如式(1)所示。

$$C_{SSD}(p, d) = \sum_{q \in N(p)} |I_l(q) - I_r(qd)|^2 \quad (1)$$

式(1)中,  $N(p)$  表示以  $p$  为中心的匹配窗口内的像素,  $q$  表示像素  $p$  的邻域像素,  $I_l(q)$  为左图像中像素  $q$  的灰度值,  $I_r(qd)$  为右图上与像素  $q$  对应视差为  $d$  的灰度值,  $C_{SSD}(p, d)$  为像素  $p$  在视差值为  $d$  处的匹配代价。

在实验与实际应用中发现,匹配窗口对匹配精度有较大的影响,匹配窗口的大小将直接影响最终的匹配精度。实验分析发现,匹配窗口太小,无法很好的表达局部区域图像的特征,易导致误匹配;窗口太大,包含冗余信息增多,不利于区分图像前景和背景,同时会增加计算量。在同一区域用不同匹配窗口在 Middlebury 数据集的 Cone 图上进行实验,结果如图2所示,右边每一行对应原图像从上到下的三个黑框区域,红色区域为误匹配标注区域,红色越少表示误匹配越少,  $w$  表示窗口的尺寸,从左分别是  $3 \times 3$ 、 $5 \times 5$  和  $9 \times 9$  的窗口得到的视差图;从图2中可以明显的发现,同一区域用不同匹配窗口处理,匹配结果不同,图中选择的三个区域为纹理丰富的边缘区域,因此适合用小窗口进行处理。

图像梯度为图像像素在  $x$  和  $y$  方向上的一阶偏导数,可以反映图像灰度的变化,可用来判别图像内容的边缘与平坦区域,在边缘处图像梯度较大,对应立体匹配中的深度不连续区域;在平坦区域图像梯度较小,对应立体匹配中弱纹理或无纹理区域,则可用梯度信息判别深度不连续区域、弱纹理或无纹理区域。定义  $x$  和  $y$  方向的梯度如式(2)与式(3)所示。

$$\begin{cases} grad_x = \partial G / \partial x \\ grad_y = \partial G / \partial y' \end{cases} \quad (2)$$

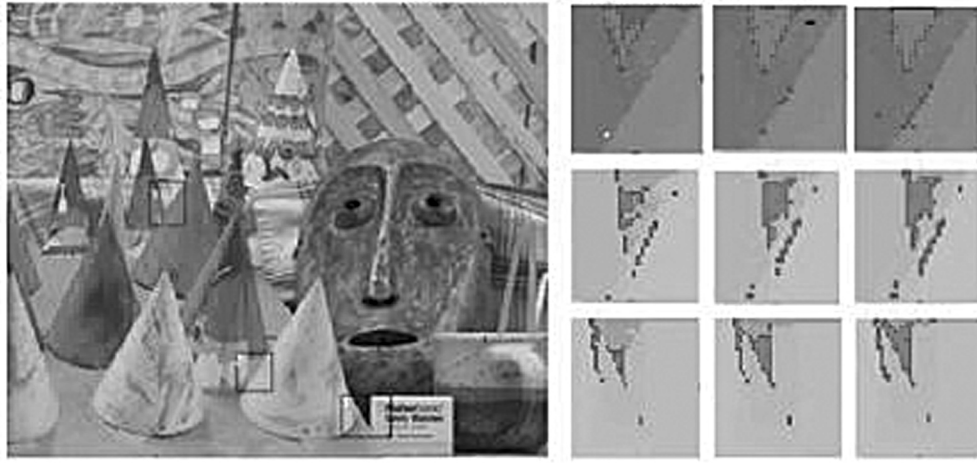


图2 Cones左图像不同匹配窗口在同一区域实验结果图

Fig. 2 Experimental results of different support windows of Cones left image in the same region

$$grad = \sqrt{(grad_x)^2 + (grad_y)^2} \quad (3)$$

式(2)和式(3)中  $G$  为图像灰度,  $grad_x$  为图像在  $x$  方向上的梯度,  $grad_y$  为图像在  $y$  方向上的梯度,  $grad$  为图像梯度, 将每个像素的梯度组合起来就产生对应图像的梯度图。

针对视差不连续区域, 本文引入自适应匹配窗口, 根据像素梯度的大小选择匹配窗口的尺寸。在深度不连续区域, 梯度值较大, 细节信息复杂, 应该选择较小的匹配窗口; 在深度连续区域, 梯度值较小, 通常是弱纹理或者无纹理区域, 细节信息较简单, 应该选择较大的窗口。根据多次实验结果与分析, 设定匹配窗口选择策略如式(4)所示。

$$\begin{cases} W = 3, grd \geq Th_{gh} \\ W = 5, Th_{gl} < grd < Th_{gh} \\ W = 9, grd \leq Th_{gl} \end{cases} \quad (4)$$

式(4)中,  $W$  表示匹配窗口大小,  $grd$  表示梯度值,  $Th_{gh}$ 、 $Th_{gl}$  分别表示最高和最低梯度的阈值。从图2中可以明显看出匹配窗口不同, 红色标记的误匹配像素数目也不同, 可以说明所提策略的有效性。

**基于梯度信息的匹配代价对光照失真以及噪声等有较好的鲁棒性**, 所以本文将基于自适应窗口的SSD代价与梯度代价进行融合得到最终的代价; 融合代价计算公式如式(5)所示。

$$C(p, d) = 2 - \exp(-C_g/T_g) - \exp(-C_{SSD}/T_{SSD}) \quad (5)$$

式(5)中,  $C_g$  与  $C_{SSD}$  分别为梯度与SSD代价,  $C(p, d)$  融合后的代价,  $T_g$  为梯度截断值, 它与边界梯度阈值设置一致,  $T_{SSD}$  为SSD代价截断值, 实验中取45, 表示灰度差值。

### 3.2 匹配代价聚合

代价计算中仅考虑了局部相关性, 无法直接进行最优视差的计算, 需要通过代价聚合来突出像素之间的相关性。常用引导滤波算法进行代价聚合, 传统引导滤波算法中局部窗口固定, 本文利用2.1章节中选择的自适应局部窗口。建立引导滤波模型如式(6)所示。

$$C'(p, d) = a_k G(p) + b_k, \forall p \in \omega_k \quad (6)$$

式(6)中,  $k$  为以  $p$  为中心窗口  $\omega_k$  内的邻域像素;  $d$  为视差;  $C'(p, d)$  为输出聚合后的匹配代价图;  $G(p)$  为输入引导图;  $a_k, b_k$  为窗口中心位于  $p$  时线性函数的系数,  $\omega_k$  为2.1章节代价计算中的选择的局部窗口。引导滤波过程可视为一个优化过程, 用以求最优解<sup>[15]</sup>; 构建的优化函数模型如式(7)所示。

$$E(a'_k, b'_k) = \sum_{k \in \omega_k} ((a'_k G(p) + b'_k - C(p))^2 + \lambda a_k'^2) \quad (7)$$

式(7)中,  $C(p)$  为输入待滤波图, 即匹配代价图;  $\lambda$  为正则化系数, 为0.0625, 对  $a'_k$  进行约束, 防止过大。利用最小二乘法求解可得最优  $a'_k, b'_k$ , 如式(8)与式(9)所示。

$$a'_k = \frac{\frac{1}{|\omega|} \sum (G(p)C(p) - \mu_k \bar{C}_k)}{\sigma_k^2 + \lambda} \quad (8)$$

$$b'_k = \bar{C}_k - a_k \mu_k \quad (9)$$

式(8)与式(9)中  $\mu_k$  和  $\sigma_k^2$  分别表示引导图像  $G(p)$  在局部窗口  $\omega_k$  中的均值和方差,  $|\omega|$  局部窗口  $\omega_k$  中的像素个数,  $\bar{C}_k$  输入图像的在窗口  $\omega_k$  中的均值。最后可得聚合代价如式(10)所示。

$$C'(p, d) = \bar{a}_k C(p) + \bar{b}_k \quad (10)$$

式中 $\overline{a'_k}$ 、 $\overline{b'_k}$ 分别为窗口 $\omega_k$ 中 $a'_k$ 、 $b'_k$ 的均值; $C'(p, d)$ 为聚合代价。

### 3.3 视差计算

传统视差计算都用 WTA(Winner-Takes-All)策略进行,选取最小代价对应的视差为最优视差,传统计算方法如式(11)所示。

$$d_{opt} = \operatorname{argmin}_{0 \leq d \leq d_{\max}} C'(p, d) \quad (11)$$

式(11)中, $d_{opt}$ 表示最小匹配代价对应的最优视差; $C'(p, d)$ 表示计算出来的代价; $d_{\max}$ 表示最大视差范围, $p$ 表示当前像素。

传统方法计算简单,容易实现,可快速获得每个像素对应的最优视差,但是该方法不能很好的消除由噪声点引发的代价突变,易得到错误的视差值;同时也不能很好的解决在弱纹理区域多个代价相同时的视差计算问题,传统方法只会随机选择一个作为视差值,这时会导致视差计算歧义问题。

针对视差计算中在弱纹理与无纹理区域存在视差歧义的问题,本文提出一种基于代价置信度约束的视差计算方法,提高了视差计算的可靠性;定义次小代价与最小代价之比为代价置信度,置信度计算

如式(12)所示。

$$R(p) = C_{\min}(p)/C_{s-\min}(p) \quad (12)$$

式(12)中 $C_{s-\min}(p)$ 表示次小代价, $C_{\min}(p)$ 表示最小代价,则代价 $R(p)$ 越小,则 $C_{\min}(p)$ 具有最小值的置信水平越高,表示该点的视差值越可靠;当置信度 $R(p)$ 大于某个阈值时,认为该点的视差值不可靠,需要对该点的视差值进行修正。视差值修正策略如式(13)所示。

$$d(p) = \begin{cases} d_{mf}(p) & R(p) > Th1 \\ d_{or}(p) & \text{otherwise} \end{cases} \quad (13)$$

式(13)中, $d(p)$ 表示最终视差, $d_{mf}(p)$ 表示中值滤波后的视差, $d_{or}(p)$ 表示初始计算的视差, $R(p)$ 表示像素 $p$ 对应的置信度, $Th1$ 表示置信度阈值。

图3为计算的视差图,红色区域为误匹配区域,其中(a)(c)添加置信度约束,(b)(d)为不添加置信度约束;从(a)~(d)误匹配率分别为17.65%、18.50%、12.78%、13.53%;可以看出(a)(c)的误匹配率小于对应(b)(d)的误匹配率,同时(a)(c)图中的红色区域少于(b)(d)图中,说明添加置信度约束可以有效提高视差图质量。



图3 加置信度与不加置信度对比图

Fig.3 Comparison of confidence and unconfidence

### 3.4 视差优化

经过前三步处理后得到初始视差图,初始视差图中包含大量由遮挡与视差不连续造成的匹配异常视差点,通常用左右一致性检测方法检测这些匹配异常点,具体做法:根据左右两幅输入图像,分别得到左右两幅视差图。对于左图中的一个像素点 $p$ ,求得最优视差值是 $d_1$ ,那么像素点 $p$ 在右图的对应点应该是像素点 $(p-d_1)$ ,像素点 $(p-d_1)$ 的视差值记为 $d_2$ ,将不满足式(14)条件的像素点 $p$ 即可视为匹配异常点, $Th2$ 表示视差阈值。

$$|d_1 - d_2| \leq Th2 \quad (14)$$

检测到匹配异常点后,需要对异常点进行修正。

传统的修正方法是用异常点水平向左和向右第一个非异常点中较小的视差来修正;该方法计算简单,由于仅选取了水平方向的异常点修正,所以会使修正区域附近出现水平条纹和边缘区曲折的现象,同时,在一些区域中仍然会存在一些不稳定点,通常会以“黑洞”的形式存在,这些不稳定点的视差值很低<sup>[16]</sup>。

为更有效的修正视差图中的异常点,本文做了大量实验,根据实验发现,这些不稳定点的视差值与其邻域的视差值都很相近,由此这里用邻域像素的视差值对视差异常点进行修正,使用水平与垂直方向交叉区域检测策略进行修正,假设像素点 $p$ 为待修正的异常视差值,修正的策略如式(15)、(16)和(17)所示。



$$dh(p) = \begin{cases} \max(d(p_l), d(p_r)), d(p_l) * d(p_r) \leq d_{th}^2 \\ \min(d(p_l), d(p_r)), d(p_l) * d(p_r) > d_{th}^2 \end{cases}, d_{th} = \rho_{th} d_{max} \quad (15)$$

$$dv(p) = \begin{cases} \max(d(p_u), d(p_d)), d(p_u) * d(p_d) \leq d_{th}^2 \\ \min(d(p_u), d(p_d)), d(p_u) * d(p_d) > d_{th}^2 \end{cases}, d_{th} = \rho_{th} d_{max} \quad (16)$$

$$d(p) = \min(dh(p), dv(p)) \quad (17)$$

式(15)、(16)和(17)中,  $d(p_l)$ 、 $d(p_r)$ 、 $d(p_u)$ 和 $d(p_d)$ 分别表示像素点  $p$  在水平方向的左、右、上和下邻近域有效点视差值;  $d_{max}$  为视差范围的最大值;  $\rho_{th}$  为判

别阈值;  $d(p)$  为最后求的视差值。实验效果如图 4 所示, 其中(a)(b)为传统算法的结果, (c)(d)为本文算法的结果; 由图 4 可以看出, 本文算法可以有效修复视差图黑洞, 进一步提高视差图的质量。

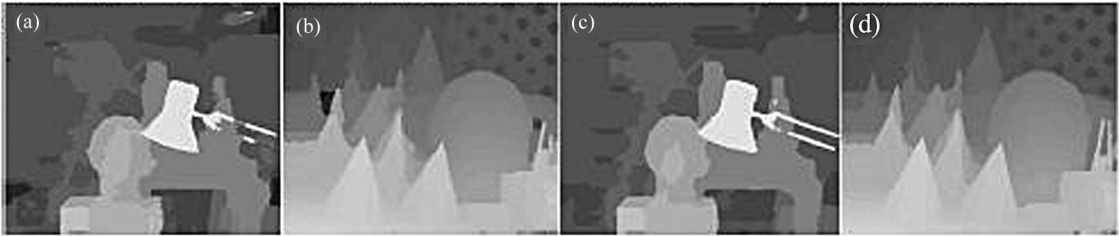


图 4 不同优化算法结果  
Fig. 4 Different optimization algorithm results

4 实验结果与分析

本文算法使用 Middlebury2.0 立体匹配评估测试平台提供的数据集进行实验, 包括 2001、2003、2005 和 2006 datasets 四个数据集中 31 组立体图像对<sup>[17]</sup>。实验环境为: Visual studio 2013 下 C/C++ 编程环境, Windows 10, X64 系统, Intel Core i7-8750H, 内存 8 GB。

为了验证所提算法的实际性能, 除非特别说明, 否则本文中涉及到的视差图都为初始视差图, 未做后处理; 对比实验中误差限为 1 个像素, 即当匹配结果与真实视差图相差一个像素以上时, 认为该点是误匹配点, 实验中涉及到的参数设置如表 1 所示。

表 1 所提立体匹配算法涉及的参数  
Tab. 1 The parameters involved in the proposed stereo matching algorithm

Parameter	Value	Parameter	Value	Parameter	Value
$OTQ2$	0.05	$\rho_{th}$	0.15	$Th_{gl}$	
$Th1$	0.8	$Th2$	1	$Th_{gh}$	0.05

4.1 代价计算验证

为验证所提算法中自适应窗口的有效性, 本文分别使用三种不同匹配窗口进行实验, 与所提算法进行对比, 通过对比视差图的误匹配率来说明算法

的有效性, 选取窗口尺寸分别为  $3 \times 3$ 、 $5 \times 5$  和  $9 \times 9$ , 考虑到需要包含丰富纹理、弱纹理以及无纹理等场景, 本次实验选择 Middlebury 数据集中不同纹理特征类型的立体图像对进行实验, 有 Teddy、Aloe、Baby1 和 Books 立体图像对。表 2 为四组实验图在所有区域的误匹配率结果对比, 图 5 为不同匹配窗口计算的视差图,  $w$  表示匹配窗口尺度, 从左分别是  $w=3$ 、5、9 和所提算法得到的视差图。

从表 2 可以看出所提算法有效的降低了误匹配率, 证明了根据梯度特征选择匹配窗口尺度策略的可行性。从图 5 可以直观看出所提算法在视差不连续与弱纹理区域中用红色标注的误匹配点少于固定匹配窗口的数量, 尤其在边缘处红标注点较稀疏; 实验说明在不同纹理区域选择不同尺度的匹配窗口可以有效降低误匹配率。

表 2 所有区域(all)不同匹配代价计算方法误匹配百分比(%)  
Tab. 2 Percentage of false match in all regions of different matching cost algorithm (%)

Algorithm	$w=3$	$w=5$	$w=9$	Proposed
Teddy	19.15	18.97	17.83	16.55
Aloe	17.03	17.44	18.50	15.54
Baby1	11.60	11.34	11.44	9.83
Books	21.06	20.90	21.26	19.36
Avg	17.21	17.065	17.26	15.32

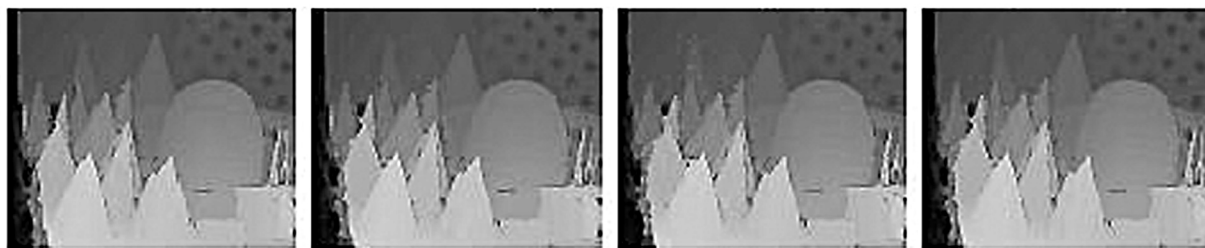


图 5 不同匹配窗口对应的视差图

Fig. 5 Disparity map corresponding to different support windows

#### 4.2 视差计算验证

为验证所提视差计算方法的有效性,这里用传统视差计算策略(WTA)同所提视差计算方法对 Middlebury 数据集中包含丰富纹理、弱纹理和无纹理场景的 6 组立体图像对进行实验,分别计算在所有区域和非遮挡区域的误匹配率。

表 3 和表 4 为不同视差计算方法在非遮挡区域和所有区域的误匹配率,从表中可以看出,所提算法在非遮挡和所有区域的平均误匹配率分别为 3.6% 和 12.73%,误匹配率低于传统算法,这是由于传统

算法不能很好的解决视差歧义问题,在视差图中存在匹配异常点,异常点越多,视差图的误匹配率越高;在视差计算阶段所提算法通过添加置信度约束,有效的解决了视差值的歧义性,所得视差图的误匹配率相对较低。图 6 为不同视差计算方法计算的视差图,图中左边红色区域为左视图边界,即在右视图中不包括的部分,所以都显示为红色标记;在其它区域中,红色标记的部分所提算法明显比传统算法少,在边缘处更明显,这是由于该部分通过置信度约束对视差异常点进行了修正,导致所提算法的精度进

表 3 非遮挡区域(no-occ)不同视差计算策略误匹配百分比(%)

Tab. 3 Percentage of false match in no-occluded region of different disparity computing algorithm (%)

Algorithm	Tsukuba	Art	Baby2	Rocks1	Cloth3	Wood2	Ave.
WTA	2.53	10.16	3.31	2.86	2.29	2.18	3.9
Proposed	2.2	9.88	3.12	2.32	2.13	2.0	3.6

表 4 所有区域(all)不同视差计算策略误匹配百分比(%)

Tab. 4 Percentage of false match in all regions of different disparity computing algorithm (%)

Algorithm	Tsukuba	Art	Baby2	Rocks1	Cloth3	Wood2	Ave.
WTA	2.86	26.44	11.96	16.21	12.03	15.54	14.71
Proposed	2.10	24.52	10.11	14.22	10.21	15.34	12.73



图 6 不同方法在 Reindeer 和 Rocks1 所有区域(all)的结果

Fig. 6 The results of different methods in all areas of Reindeer and Rocks1

一步提升。

4.3 算法对比

为验证所提优化策略的有效性,以及所提算法的整体性能,这里将对 Middlebury2.0 中所有 31 组立体图像对进行实验。

在主观分析中,对四组标准图像对进行对比实验,对比算法为 GRD<sup>[18]</sup>和 MCT'<sup>[19]</sup>,实验结果如图 7 所示,红色标注点为对应视差图中非遮挡区域的误匹配标注。从图 7 中可以看出所提算法在 Tsukuba 图像中对边缘的处理效果不如 GRD 算法,但在最终视差图里所提算法中小黑洞区域是最少的;在其它图中,所提算法对应的红色误匹配标注点明显少于

其它算法,说明所提算法有更低的误匹配率,视差图质量更高。

为便于客观分析算法性能,选择 5 种经典算法进行对比实验,对比算法包括 MCT、GC-occ<sup>[5]</sup>、SemiGlob<sup>[20]</sup>、AdaptAggrDP<sup>[21]</sup>和 RTCensus<sup>[22]</sup>,这里计算在非遮挡区域和所有区域的误匹配率,且这里的视差图都经过视差优化算法处理。从表 5 中可以看出,所提算法的精度得到进一步提升,在所有算法中平均误匹配率最低,分析实验结果可得,所提算法在边缘细节方面的精度得到进一步提升,同时视差图中的小黑洞得到修正,明显少于其它算法,视差图质量更高。

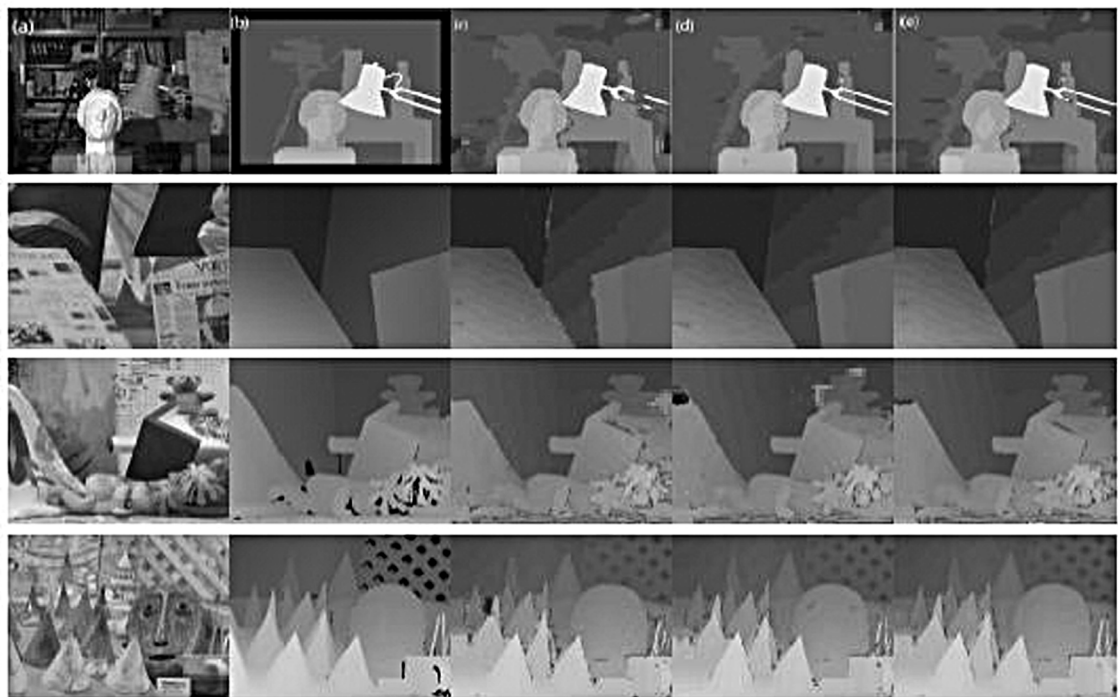


图 7 实验结果图

Fig. 7 Experimental results

表 5 不同立体匹配算法在 4 组标准图像上的误匹配率(%)

Tab. 5 Rate of false match between different stereo matching algorithms on 4 sets of standard images (%)

Algorithm	Tsukuba		Venus		Teddy		Cones		Avg
	no-occ	all	no-occ	all	no-occ	all	no-occ	all	
MCT'	4.01	4.59	1.02	1.64	8.93	16.1	5.19	14.29	6.97
GC-occ	1.19	2.01	1.64	2.19	11.20	17.40	5.36	12.4	6.67
AdaptAggrDP	1.57	3.50	1.53	2.69	6.79	14.3	5.53	13.2	6.14
RTCensus	5.08	6.25	1.58	2.42	7.96	13.8	4.10	9.54	6.34
SemiGlob	3.26	3.96	1.00	1.57	6.02	12.2	3.06	9.75	5.10
Proposed	2.14	3.05	0.33	0.97	5.30	10.9	3.00	8.67	<b>4.28</b>

为充分说明算法的可靠性,在 Middlebury2.0 数据集集中的 31 组立体图像对进行实验,实验结果如表 6 所示,这里将平均误匹配率作为对比指标,对比算法包括 MCT、MCT'<sup>[19]</sup>、MST<sup>[6]</sup>、AW<sup>[23]</sup>、GF<sup>[24]</sup>、CT-GF<sup>[18]</sup>和 CT-MST<sup>[18]</sup>。这里的视差图经过视差优化处理,CT-GF、CT-MST 和所提算法都是在三层金字塔下处理。

从表 6 中可以看到,所提算法在所有算法中平均误匹配率最低,为 7.96%,MCT 算法的平均误匹配率最高;实验分析发现,所提算法在边缘处的误匹

配像素分布比其它算法稀疏(如图 6),说明所提算法可以减少边缘处的误匹配点,这是由于使用了自适应窗口策略,同时添加了置信度约束的原因;同时发现在最终视差图中的小黑洞也明显少于其它算法,这是由于所提算法在视差优化中进行了交叉验证的视差优化处理;多次实验得出,连续两次优化效果最合适,这样可以平衡精度和效率的性能;第二次优化是在第一次优化后的基础上进行,进一步提升了算法的综合性能。相比其它算法,所提算法得到的视差图质量较高。

表 6 不同算法在 31 组立体图像中的平均误匹配百分比(%)

Tab. 6 Average false match percentage of different algorithms in 31 stereo images(%)

Algorithm	MCT	MCT'	MST	AW	GF	CT-GF	CT-MST	SWCT	Proposed
Avg.	14.33	12.63	10.42	11.95	10.02	8.89	10.11	8.35	7.96

## 4 结 论

本文提出一种融合梯度特性与置信度的立体匹配算法。在匹配代价计算中加入自适应窗口,很好的适应了不同纹理区域,在不同纹理区域使用不同窗口,有效的降低了误匹配率;为计算视差值添加置信度约束条件,有效解决了视差计算中的视差歧义问题;在视差优化中,使用邻域视差值修复黑洞视差区域,通过对比实验分析,所提算法的策略得到有效的验证,视差图的误匹配率进一步降低。但是在分析中发现,本文算法在边缘保持上不如基于梯度的算法,同时在遮挡区域还需要更进一步提升。今后研究中,将对每一项指标进行更加深入的分析研究,尤其在匹配代价计算与代价聚合中。

## 参考文献:

- [1] LIU Huan. Research and application of stereo matching algorithm based on binocular vision [D]. Heilongjiang: Harbin University of Technology, 2018.  
刘欢. 基于双目视觉立体匹配算法的研究与应用[D]. 黑龙江: 哈尔滨工业大学, 2018.
- [2] XU Huo-xi, Jiang Ke, Fang Hao. Color image stereo matching algorithm using neighborhood weights and color information[J]. Journal of Optoelectronics • Laser, 2015, 26(10): 2031-2036.  
徐火希, 姜柯, 方浩. 结合邻域权值和色彩信息的彩色立体匹配算法[J]. 光电子 • 激光, 2015, 26(10): 2031-

- 2036.
- [3] Scharstein D, Szeliski R, Zabih R. A Taxonomy and evaluation of dense two-frame stereo correspondence algorithms[C]. SMBV, IEEE Computer Society, 2001, 0131.
- [4] Pedro F Felzenszwalb, Daniel P. Huttenlocher. Efficient belief propagation for early vision[J]. International Journal of Computer Vision, 2006, 70(1): 41-54.
- [5] Kolmogorov V, Zabih R. Computing visual correspondence with occlusions using graph cuts[C]. ICCV, 2001, 508-515.
- [6] Yang Q. A non-local cost aggregation method for stereo matching[C]. CVPR, 2012, 1402-1409.
- [7] Yao P, ZHANG H, XUE Y, et al. Iterative color-depth MST cost aggregation for stereo matching[C]. IEEE International Conference on Multimedia and Expo. IEEE, 2016, 1-6.
- [8] Mei X, Sun X, Dong W, et al. Segment-tree based cost aggregation for stereo matching[C]. Computer Vision and Pattern Recognition. IEEE, 2013, 313-320.
- [9] Jurežbontar, YANN Lecun. Stereo matching by training a convolutional neural network to compare image patches [C]. 2015, CVPR. 1592-1599.
- [10] Xiao J S, Tian H, Zou W T, et al. Stereo matching based on convolutional neural network[J]. Acta Optica Sinica, 2018, 38(8): 0815017.  
肖进胜, 田红, 邹文涛, 等. 基于深度卷积神经网络的双目立体视觉匹配算法[J]. 光学学报, 2018, 38(8):



0815017.

- [11] Mayer N, ILGg E, HÄUSSER Philip, et al. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation[C]. CVPR, 2015, 16526849.
- [12] Pang J, Sun W, Ren J S, et al. Cascade residual learning: a two-stage convolutional neural network for stereo matching[C]. ICCV, 2017, 978:887-895.
- [13] FEDERICO Tombari, STEFANO Mattoccia, STEFANO L D, et al. Classification and evaluation of cost aggregation methods for stereo correspondence[C]. CVPR, 2008, 1-8.
- [14] Bleyer M, Rhemann C, Rother C. Patch match stereo-stereo matching with slanted support windows[C]. BMVC, 2013, 1-11.
- [15] Fan H R, Yang F, Pan X R, et al. Stereo matching algorithm for improved census transform and gradient fusion [J]. Acta Optica Sinica, 2018, 38(2): 0215006.  
范海瑞, 杨帆, 潘旭冉, 等. 一种改进 Census 变换与梯度融合的立体匹配算法[J]. 光学学报, 2018, 38(2): 0215006.
- [16] JIAO Jian-bo, WANG Rong-gang, WANG Wen-ming, et al. Local stereo matching with improved matching cost and disparity refinement[J]. IEEE Multimedia, 2014, 21(4): 16-27.
- [17] Scharstein D, Szeliski R. The middle stereo vision page. <http://vision.middlebury.edu>. 2018.
- [18] Zhang K, Fang Y, Min D, et al. Cross-scale cost aggregation for stereo matching[C]. CVPR, 2014, 1590-1597.
- [19] ZHU Shi-ping, YAN Li-na, LI Zheng. Stereo matching algorithm based on improved census transform and dynamic programming [J]. Acta Optica Sinica, 2016, (4): 0415001.  
祝世平, 闫利那, 李正. 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, (4): 0415001.
- [20] Hirschmuller H. Accurate and efficient stereo processing by semi-global matching and mutual information [C]. CVPR, 2005, 807-814.
- [21] WANG Liang, YANG Rui-gang, GONG Ming-lun, et al. Real-time stereo using approximated joint bilateral filtering and dynamic programming[J]. Journal of Real-Time Image Processing, 2014, 9(3): 447-461.
- [22] Martin Humenberger, Zinner Christian, Michael weber, et al. A fast stereo matching algorithm suitable for embedded real-time systems[J]. Computer Vision & Image Understanding, 2010, 114(11): 1180-1202.
- [23] YOON Kuk-jin, KWEON In-so. Adaptive support-weight approach for correspondence search[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2006, 28(4): 650-656.
- [24] Hosni A, Rhemann C, Bleyer M, et al. Fast cost-volume filtering for visual correspondence and beyond[J]. TPAMI, 2013, 35(2): 504-11.

#### 作者简介:

杨帆 (1966—), 男, 天津, 博士, 河北工业大学电子信息工程学院, 教授, 博士生导师, 主要从事智能信息、图像处理与机器视觉方面的研究。