

# 改进的跨尺度引导滤波立体匹配算法<sup>①</sup>

杜晨瑞, 李英祥

(成都信息工程大学 通信工程学院, 成都 610225)

通讯作者: 杜晨瑞, E-mail: [string\\_dcr@163.com](mailto:string_dcr@163.com)



**摘 要:** 作为双目三维重建中的关键步骤, 双目立体匹配算法完成了从平面视觉到立体视觉的转化. 但如何平衡双目立体匹配算法的运行速度和精度仍然是一个棘手的问题. 本文针对现有的局部立体匹配算法在弱纹理、深度不连续等特定区域匹配精度低的问题, 并同时考虑到算法实时性, 提出了一种改进的跨多尺度引导滤波的立体匹配算法. 首先融合 AD 和 Census 变换两种代价计算方法, 然后采用基于跨尺度的引导滤波进行代价聚合, 在进行视差计算时通过制定一个判断准则判断图像中每一个像素点的最小聚合代价对应的视差值是否可靠, 当判断对应的视差值不可靠时, 对像素点构建基于梯度相似性的自适应窗口, 并基于自适应窗口修正该像素点对应的视差值. 最后通过视差精化得到最终的视差图. 在 Middlebury 测试平台上对标准立体图像对的实验结果表明, 与传统基于引导滤波器的立体匹配算法相比具有更高的精度.

**关键词:** 立体匹配; 匹配代价; 多尺度; 引导滤波; 自适应窗口

引用格式: 杜晨瑞, 李英祥. 改进的跨尺度引导滤波立体匹配算法. 计算机系统应用, 2019, 28(4): 176–182. <http://www.c-s-a.org.cn/1003-3254/6868.html>

## Improved Stereo Matching Algorithm Based on Cross-Scale Guided Filtering

DU Chen-Rui, LI Ying-Xiang

(School of Communication Engineering, Chengdu University of Information Technology, Chengdu 610225, China)

**Abstract:** As a key step in binocular 3D reconstruction, the binocular stereo matching algorithm completes the transformation from planar vision to stereo vision. But how to balance the running speed and accuracy of the binocular stereo matching algorithm is still a difficult problem. In this study, focused on that existing local stereo matching algorithm has low matching accuracy in specific regions such as weak texture and depth discontinuity, while considered the real-time performance of the algorithm at the same time, an improved stereo matching algorithm based on cross-scale guided filtering is proposed. Firstly, the two cost calculation methods of SAD and Census transform are combined, and then the cost aggregation is performed by using cross-scale guided filtering. When calculating the disparity calculation, a judgment criterion is used to judge the reliance of the disparity value corresponding to the minimum aggregation cost of each pixel in the image. When it is judged that the corresponding disparity value is unreliable, an adaptive window based on gradient similarity is constructed for the pixel, and the disparity value corresponding to the pixel is corrected based on the adaptive window. Finally, the final disparity map is obtained by parallax refinement. Experimental results on standard stereo image pairs on the Middlebury test platform show higher accuracy than traditional guided filter based stereo matching algorithms.

**Key words:** stereo matching; matching cost; multi-scale; guided filtering; adaptive window

① 收稿时间: 2018-11-01; 修改时间: 2018-11-23; 采用时间: 2018-11-27; csa 在线出版时间: 2019-03-28

近年来随着计算机视觉领域的迅猛发展, 基于视觉的三维重建已经成为一个非常热门的方向, 双目视觉作为其中一种重要的分支也越来越得到大家的关注. 立体匹配是从双目视觉到三维重建中的一个非常关键的步骤, 对重建效果影响巨大. 根据 Sharstein 等<sup>[1]</sup>提出的双目立体匹配理论框架, 立体匹配主要分为以下 4 个步骤: 匹配代价计算、匹配代价聚合、视差计算以及视差精化. 当前主要的立体匹配算法又可以根据在匹配代价计算后是否采用代价聚合分为全局和局部立体匹配算法. 全局立体匹配算法的核心思路是在计算匹配代价后, 构建一个全局的能量函数, 通过全局优化理论方法最小化能量函数估计视差. 全局立体匹配算法匹配精度高, 但其运行速度慢且算法复杂度较高. 局部立体匹配算法充分利用像素点周围的局部信息, 通过代价聚合来估计视差, 得到视差的局部最优解. 相比于全局立体匹配算法, 精度稍低, 但运行速度快且算法复杂度较低.

现有的局部立体匹配算法主要关注于改进代价计算和代价聚合两部分来提高算法精度. 常见的匹配代价计算方法有基于像素的匹配代价如灰度绝对值 (Absolute Differences, AD)、灰度差平方 (Squared Differences, SD) 等; 基于窗口的匹配代价如灰度差绝对值和 (Sum of Absolute Difference, SAD)、归一化互相关 (Normalized Cross Correlation, NCC) 等; 以及基于非参数变换<sup>[2]</sup>的匹配代价如 Census、Rank 等. 一般来说, 基于像素和窗口的代价计算方法对噪声和光照变化非常敏感, Census 变换对光照变化不敏感, 但在重复纹理区域效果较差, 因此融合多种代价计算方法也成为一种常用的手段. Hosni 等将 SAD 和梯度信息结合作为性的代价计算方法; Mei 等<sup>[3]</sup>结合灰度差绝对值和 Census 变换进行代价计算, 在弱纹理区域和重复纹理结果区域都取得了很好的结果; Zhu 等<sup>[4]</sup>提出将 AD、梯度信息和 Census 变换结合, 提高了匹配精度.

代价聚合的方法主要分为自适应窗口和自适应权重的方法. Yoon 等<sup>[5]</sup>首先提出使用双边滤波器进行代价聚合, 提出了自适应权重的算法, 该算法的主要思想视为匹配窗口中每一个像素赋予一个权重, 而权重通过其到窗口中心点的颜色差和距离得到, 但算法复杂度较高, 且运行速度较慢. Hosni 等<sup>[6]</sup>提出使用引导滤波替代双边滤波进行代价聚合, 该算法整个计算过程都是线性的, 计算效率大幅提高, 在保持运行速度的情况下具有良好的匹配精度. Zhu 等<sup>[7]</sup>优化 Census 变换策略, 并讲动态规划引入到代价聚合中, 有效的提升了代价聚合的效率.

为在保持运行速度的前提下, 提高视差精度, 本文

提出了一种改进的基于跨尺度引导滤波的立体匹配算法. 首先融合 SAD 和改进的 Census 变换两种代价计算方法, 然后采用基于跨尺度的引导滤波进行代价聚合, 在进行视差计算时, 并没有采用常用的赢者通吃 (Winner-Take-All, WTA) 方法, 而是通过制定一个判断准则判断图像中每一个像素点的最小聚合代价对应的视差值是否可靠, 当判断为视差不可靠时, 对像素点构建自适应十字交叉窗口, 并基于自适应窗口修正该像素点对应的视差值, 最后进行视差精化得到最终的视差图. 实验表明, 相比于当前的局部立体匹配算法, 本文所提出的算法可以获得更高精度的视差图.

## 1 引导滤波

由 He 等<sup>[8]</sup>提出的引导滤波算法, 是一种能够实现边缘保留特性的滤波方法.

设  $I$  为引导图像,  $p$  为待滤波图像,  $q$  为滤波后图像, 引导滤波基于如下假设: 在以各个像素为中心的窗口  $\omega_k$  内,  $q$  由  $I$  线性转换得到:

$$q_i = a_k I_i + b_k, \forall i \in \omega_k \quad (1)$$

$$q_i = p_i - n_i, \forall i \in \omega_k \quad (2)$$

式 (1) 中,  $\omega_k$  表示以像素  $k$  为中心、以  $r$  为半径的方形窗口;  $a_k$  和  $b_k$  表示线性系数, 在窗口  $\omega_k$  内为固定值.

在窗口  $\omega_k$  内, 通过最小化噪声能量求解线性参数  $a_k$  和  $b_k$ :

$$E(a_k, b_k) = \sum_{i \in \omega_k} [(a_k I_i + b_k - p_i)^2 + \varepsilon a_k^2] \quad (3)$$

式 (3) 中,  $\varepsilon$  表示一个正则化系数.  $\varepsilon a_k^2$  作为平滑项, 防止线性系数  $a_k$  过大, 通常有  $\varepsilon > 0$ . 正则化系数  $\varepsilon$  对线性系数  $a_k$  有惩罚作用, 当  $\varepsilon$  取值越大, 惩罚作用越强, 滤波后图像  $q$  越平滑; 当  $\varepsilon$  取值越小, 惩罚作用越弱, 滤波后图像  $q$  边缘保留越完整.

式 (3) 的解为:

$$a_k = \frac{\frac{1}{|\omega|} \sum_{i \in \omega_k} I_i p_i - \mu_k p_k}{\sigma_k^2 + \varepsilon} \quad (4)$$

$$b_k = p_k - a_k \mu_k \quad (5)$$

式 (4)、(5) 中,  $\mu_k$  和  $\sigma_k^2$  分别表示引导图像  $I$  在窗口  $\omega_k$  内的均值和方差;  $|\omega|$  表示窗口  $\omega_k$  内的像素数;  $p_k$  表示待滤波图像  $p$  在窗口  $\omega_k$  内的均值.

## 2 本文算法

本文算法由代价计算、跨尺度引导图滤波以及视差计算三部分组成. 核心算法流程如图 1 所示. 首先对原始

图像进行降采样,得到多尺度图像,并对每一尺度图像对计算其匹配代价,构建多尺度原始代价空间;然后基于引导滤波对多尺度原始代价空间进行代价聚合,并加入正则式项,优化相邻尺度的一致性;最后以处理后的代价空间为基础计算视差,根据视差判断准则,对不可靠视差进行优化。

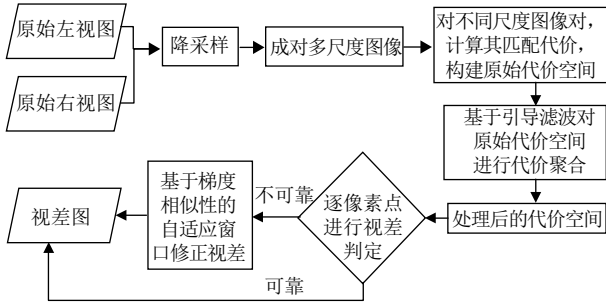


图1 算法流程图

## 2.1 代价计算

代价计算的目的是量化左右图像对应像素之间的相似程度,在设计代价计算方法时面对的主要问题是像素级的相似性判断对光照变化、噪声等非常敏感,非常容易误匹配。一般的计算方法有 SAD、Census 和归一化互相关等,同时多种代价计算方法融合也是很常用的方法。本文采用的基于颜色通道的 SAD 和 census 融合的代价计算方法。

假设左图中的一个像素点为  $p_i$ , 坐标为  $(x, y)$ , 相对应的右图中的像素点为坐标为  $(x-d, y)$ , 其中  $d$  为视差。以像素点  $p_i$  为中心, 半径为  $r$  的矩形窗位  $\omega_i$ , 分别计算  $C_{AD}(p_i, d)$  和  $C_{census}(p_i, d)$  的值。

$$C_{AD}(p_i, d) = \frac{1}{3} \sum_{c \in [r, g, b]} I_l^c(\omega_i) - I_r^c(\omega_i) \quad (6)$$

$$I(\omega_i) = \frac{1}{N} \sum_{j \in \omega_i} p_j \quad (7)$$

式 (6)、(7) 中  $I(\omega_i)$  表示在矩形框  $\omega_i$  中, 包含的像素点的平均值,  $N$  为矩形框的像素点个数。  $I_l^c(\omega_i)$  和  $I_r^c(\omega_i)$  用于区分单一颜色通道和左右图像。

在计算  $C_{census}(p_i, d)$  时, 采用在左图像对应的灰度图像中计算矩形窗内像素的平均值, 并依次比较矩形窗中像素点与平均值, 得到二进制编码  $cen_l(p_i)$ , 然后根据同样的方法计算右图的  $cen_r(p_i)$ 。通过计算两个二进制编码串的 Hamming 距离, 得到  $C_{census}(p_i, d)$ 。其计算公式如下:

$$C_{census}(p_i, d) = Ham(cen_l(p_i), cen_r(p_i)) \quad (8)$$

SAD-census 联合匹配代价的定义为:

$$C(p_i, d) = \rho(C_{census}(p_i, d), \lambda_{census}) + \rho(C_{AD}(p_i, d), \lambda_{AD}) \quad (9)$$

其中,  $\rho(C, \lambda)$  是关于  $C$  的函数:

$$\rho(C, \lambda) = 1 - \exp\left(-\frac{C}{\lambda}\right) \quad (10)$$

其中,  $\lambda_{census}$  和  $\lambda_{AD}$  参数, 其主要目的是通过该函数将两类代价值控制在  $[0, 1]$ , 使代价不会因为某一项发生较大的突变。

## 2.2 跨尺度引导图像滤波

传统的匹配代价聚合方法主要考虑原始图像, 即在最细的尺度进行研究。其聚合代价可以视为一个加权最小二乘问题<sup>[9]</sup>, 即

$$\tilde{C}(i, d) = \argmin_z \frac{1}{Z_i} \sum_{j \in \omega_i} K(i, j) z - C(j, d)^2 \quad (11)$$

式 (11) 中  $\tilde{C}(i, d)$  为中心像素  $p_i$  在视差为  $d$  时的聚合代价,  $K(i, j)$  为其邻域像素  $p_j$  的支持权重,  $Z_i = \sum_{j \in \omega_i} K(i, j)$  为归一化常数,  $\omega_i$  为中心像素  $i$  的支持窗口。 $z$  为待计算的聚合代价。

为了融合多尺度的信息, 充分利用粗尺度信息提高弱纹理和无纹理区域的匹配精度, Zhang 等<sup>[10]</sup>通过对图像下采样获取多尺度信息, 提出了跨尺度模型来聚合多尺度匹配代价, 在优化目标中添加广义 L2 正则化项, 使匹配代价在相邻尺度上具有一致性, 即:

$$\tilde{C}(i^s, d^s) = \argmin_{z^s} \left( \sum_{s=0}^S \frac{1}{Z_i^s} \sum_{j^s \in N_i^s} K(i^s, j^s) z^s - C^s(j^s, d^s)^2 + \gamma \sum_{s=1}^S z^s - z^{s-1} \right) \quad (12)$$

式 (12) 中上标  $s$  为下采样尺度空间,  $S$  为下采样的最大次数。  $\gamma$  为正则化因子。

## 2.3 视差计算

代价聚合得到的代价空间, 同时包含了对应像素之间的相似度信息和对应像素邻域之间的相似度信息。在计算每个像素点的视差值时, 传统算法一般采用赢家通吃 (WTA) 的策略简单地选择最小匹配代价对应的视差值作为该像素对应的视差值。

然而, 考虑到最小匹配代价对应的视差值不一定等于真实视差值。WTA 策略并不是一个好的选择视差的策略。根据聚合后的代价空间信息, 可以对视差的可靠

靠性进行判断. 一个常用的视差可靠的判断依据<sup>[11]</sup>如下:

$$C_d = \frac{C_{i,1}}{C_{i,2}} \quad (13)$$

$$C_i = \begin{cases} C_{i,1}, C_d \leq C_{th} \\ g(C(i,d)), C_d > C_{th} \end{cases} \quad (14)$$

式(13)、(14)中对于任意像素 $p_i$ ,  $C_{i,1}$ 表示最小匹配代价,  $C_{i,2}$ 表示次最小匹配代价.  $C_d$ 越小, 则在一定程度上说明匹配的视差值越可靠. 当设定一个阈值 $C_{th}$ , 可以和容易判断当前像素点的最小的匹配代价对应的视差值是否可靠.

对于不可靠的视差值还需要进一步的处理. 一般来说某个区域梯度变化的不规则通常会导致其和噪声具有高相似性, 从而导致引导滤波将其误认噪声. 所以本文采用基于梯度相似性的自适应窗口来处理不可靠视差值, 如图2所示.

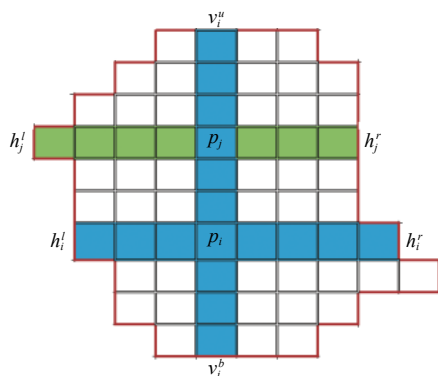


图2 基于梯度相似性的自适应窗口

从图像左上角开始, 按照从上到下、从左到右的顺序搜索视差不可靠像素点, 对于搜索到的每一个像素点 $p_i = (x_i, y_i)$ , 以像素 $p_i$ 为中心从垂直和水平两个方法构建自适应窗口. 根据图像的区域梯度结构自适应地确定臂长, 令四个方向上的臂长分别为 $h_i^l$ 、 $h_i^r$ 、 $v_i^l$ 和 $v_i^r$ . 以 $h_i^r$ 为例, 使用 $x$ 方向上的Sobel算子对像素 $i$ 右侧的像素进行梯度相似性检测, 其目的是在相邻像素梯度相似的情况下找到最大的水平右臂长. 设像素 $p_i$ 的水平右像素点为 $p_{i+j} = (x_i + j, y_i)$ ,  $j = 1, 2, 3, \dots, N$ , 则有:

$$h_i^r = \max \left[ r \prod_{\substack{j \in [1, 2, 3, \dots, N], \\ x_j \in [x_i + 1, x_i + r], \\ y_j = y_i, \\ r \in [1, L]}} g(j-1, j) \right] \quad (15)$$

$$g(j-1, j) = \begin{cases} 1, \frac{1}{3} \sum_{c \in [r, g, b]} |G_x^c(j-1) - G_x^c(j)| \leq \theta \\ 0, \text{ other} \end{cases} \quad (16)$$

式(15)、(16)中,  $L$ 表示最大臂长;  $G_x(j)$ 表示像素点 $p_{i+j}$ 在 $x$ 轴方向上的梯度值;  $\theta$ 表示梯度相似性的判定阈值.

确定 $h_i^l$ 、 $h_i^r$ 、 $v_i^l$ 和 $v_i^r$ 四个方向上的臂长后, 水平部分 $H_i$ 和垂直部分 $V_i$ 定义如下:

$$H_i = \{(x, y) | x \in [x_i - h_i^l, x_i + h_i^r], y = y_i\} \quad (17)$$

$$V_i = \{(x, y) | y \in [y_i - v_i^l, y_i + v_i^r]\} \quad (18)$$

像素点 $p_i$ 的自适应窗口 $U_i$ 定义如下:

$$U_i = \bigcup_{j \in V_i} H_j \quad (19)$$

由于自适应窗口 $U_i$ 中的像素梯度结构相似, 可以认为它们是位于相同的区域内, 所以具有相同的视差值. 选择窗口 $U_i$ 中的像素匹配代价和最小的视差值, 作为窗口 $U_i$ 中的每个视差不可靠像素的结果时差之, 即

$$g(C(i, d)) = \argmin_{j \in U_i} C_{j, l} \quad (20)$$

处理完成后, 将窗口 $U_i$ 中的所有视差不可靠像素标记为可靠像素, 继续搜索下一个视差不可靠像素进行处理, 直到所有视差不可靠的像素点全部处理.

### 3 实验结果与分析

为验证本文算法的有效性, 在Clion集成开发环境中结合图像处理开源库OpenCV, 使用C++实现了算法. 实验环境为: Ubuntu16.04系统, Intel i7-4720HQ, 内存8GB. 采用学术界公认的立体匹配算法测试平台Middlebury<sup>[12]</sup>进行测评. 该网站提供了4组标准彩色立体匹配图以及其对应的真实视差图, 分别为: Tsukuba(384像素×288像素), Venus(434像素×383像素), Teddy(450像素×375像素), Cones(450像素×375像素). 通过将实验结果视差图与真实视差图比较并量化匹配误差, 从而客观地对算法效果进行评价. 其中实验中参数设置如下:  $C_{th} = 0.7$ ,  $\theta = 1.0$ .

#### 3.1 结果与分析

图3中为在Middlebury2006标准立体图像数据集上的实验结果. 图3(a)为各组标准图像对的原始左图像, 图3(b)为各组标准图像对的真实视差图, 图3(c)为单尺度引导滤波算法(GF)<sup>[6]</sup>的视差图, 图3(d)为跨尺



度引导滤波算法 (S-GF)<sup>[10]</sup> 的视差图, 图 3(e) 为本文算法的视差图。以上几种算法与本文算法相关性较大, 具有很高的可比性。由对比结果可以看出, 本文算法对 4 对标准图像对都有一定程度的精度提升。

表 1 为本文算法与其他局部立体匹配算法的定量结果对比。其中 n-occ 为非遮挡区域误匹配像素百分比, disc 为深度不连续区域误匹配像素百分比, all 是全部区域误匹配像素百分比。通过它们的均值可以得出 4 组标准图像对的平均错误率 (Average Percent of Bad Pixels, Avg PBM)。从表 1 数据可以看出, 本文算法同 GF 算法相比精度提高了 0.81 个百分点, 同 S-GF 算法相比提高了 0.2 个百分点。本文算法尽管在 Tsukuba 精度上有所降低, 但在 Venus, Teddy 和 Cones 的匹配精度上有了显著提高。

### 3.2 参数分析

参数  $C_{th}$  为判断像素点视差是否可靠的判定阈值,

参数  $\theta$  为构建基于梯度的自适应十字交叉窗口时, 判断梯度结构相似性的阈值。这两个参数对视差计算得到视差图的精度都起到了非常大的影响。对于参数  $C_{th}$  而言, 其越小判断像素点视差值是否可靠的标准越严格, 越大则判断标准越放松。对于参数  $\theta$  而言, 其越小则构建的十字交叉窗口越小, 越大则构建的十字交叉窗口越大。十字交叉窗口的大小会直接影响匹配精度。为得到最高的匹配精度, 需要通过实验确定合理的  $\theta$  值。为得到最高的匹配精度, 需要通过实验确定合理的  $C_{th}$  和  $\theta$  值。在保持其他实验条件和参数不变的情况下, 对  $C_{th}$  从 [0.3, 1] 之间的取值进行了实验, 计算匹配误差, 实验结果如图 4 所示。在保持其他实验条件和参数不变的情况下, 对  $\theta$  从 [0.2, 2] 之间的取值进行了实验, 计算匹配误差, 实验结果如图 5 所示。

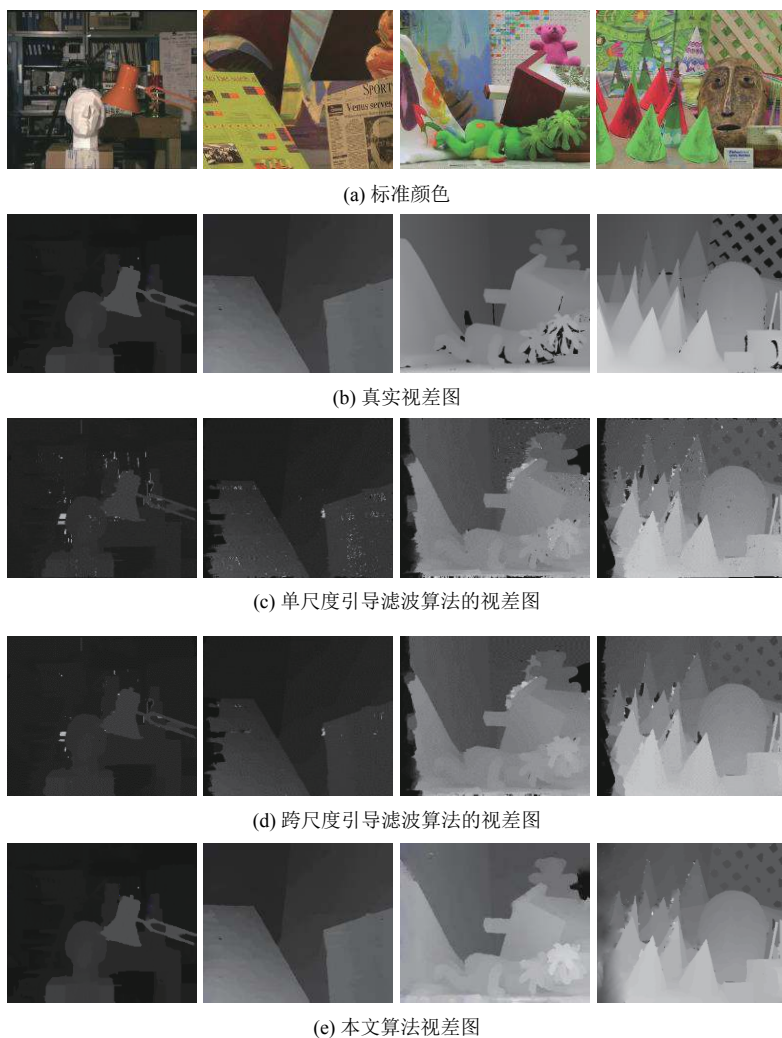


图 3 标准图像对不同方法下的视差图对比

表 1 不同算法误匹配像素百分比

算法	Tsukuba			Venus			Teddy			Cones			AvgPBM
	n-occ	disc	all	n-occ	disc	all	n-occ	disc	all	n-occ	disc	all	
GF	2.62	8.16	3.38	1.72	13.2	2.81	16.5	3.24	17.5	3.34	8.70	12.2	8.10
S-GF	3.20	8.12	2.84	1.10	9.67	2.00	7.04	16.7	15.1	3.00	8.17	11.4	7.49
本文算法	2.38	8.5	2.97	1.13	9.26	1.99	7.05	16.8	14.9	3.32	7.99	11.0	7.29

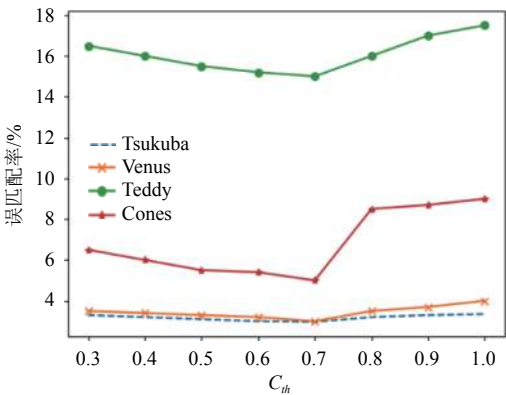


图 4 不同 $C_{th}$ 取值下的误匹配率

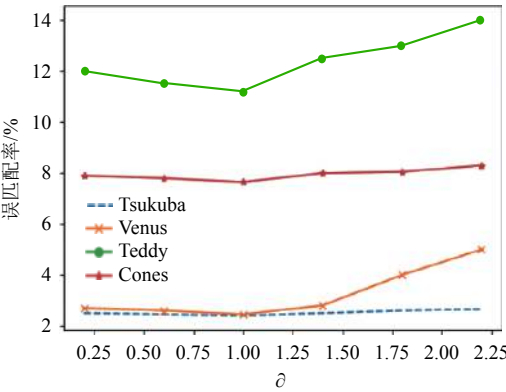


图 5 不同 $\delta$ 取值下的误匹配率

从图 4、图 5 可以看出, 对于 Venus 数据, 误匹配率受参数 $C_{th}$ 和 $\delta$ 值得影响不明显. 而对于其他 3 组图像数据, 当 $C_{th} = 0.7$ ,  $\delta = 1.00$ 时, 得到的视差图精度最好.

4 结论

为提高视差图精度同时保持较快的运行速度, 本文提出了一种改进的跨尺度引导滤波算法. 首先融合 SAD 和 Census 变换两种代价计算方法, 然后采用基于跨尺度的引导滤波进行代价聚合, 在进行视差计算时, 并没有采用常用的赢者通吃 (Winner-Take-All, WTA) 方法, 而是通过制定一个判断准则判断图像中每一个像素点的最小聚合代价对应的视差值是否可靠, 当判断为视差不可靠时, 对像素点构建基于梯度相似

性的自适应窗口, 并基于自适应窗口修正该像素点对应的视差值, 最后进行视差精化得到最终的视差图. 实验结果表明, 通过达到精度和速度的平衡. 在接下来的研究中, 会基于 GPU 平台重构代码, 实现并行运算进一步提高算法运行速度.

参考文献

- 1 Scharstein D, Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision, 2002, 47(1): 7–42.
- 2 Zabih R, Woodfill J. Non-parametric local transforms for computing visual correspondence. Proceedings of the third European Conference on Computer Vision (Vol. II). Stockholm, Sweden. 1994. 151–158.
- 3 Mei X, Sun X, Zhou MC, et al. On building an accurate stereo matching system on graphics hardware. Proceedings of 2011 IEEE International Conference on Computer Vision Workshops. Barcelona, Spain. 2011. 467–474.
- 4 Zhu SP, Yan L. Local stereo matching algorithm with efficient matching cost and adaptive guided image filter. The Visual Computer, 2017, 33(9): 1087–1102. [doi: 10.1007/s00371-016-1264-6]
- 5 Yoon KJ, Kweon IS. Adaptive support-weight approach for correspondence search. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(4): 650–656. [doi: 10.1109/TPAMI.2006.70]
- 6 Hosni A, Rhemann C, Bleyer M, et al. Fast cost-volume filtering for visual correspondence and beyond. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(2): 504–511. [doi: 10.1109/TPAMI.2012.156]
- 7 Zhu SP, Yan LN, Li Z. Stereo matching algorithm based on improved Census transform and dynamic programming. Acta Optica Sinica, 2016, 36(4): 0415001.
- 8 He KM, Sun J, Tang XO. Guided image filtering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(6): 1397–1409. [doi: 10.1109/TPAMI.2012.213]
- 9 Milanfar P. A tour of modern image filtering: New insights and methods, both practical and theoretical. IEEE Signal

- Processing Magazine, 2013, 30(1): 106–128. [doi: [10.1109/MSP.2011.2179329](https://doi.org/10.1109/MSP.2011.2179329)]
- 10 Zhang K, Fang YQ, Min DB, *et al.* Cross-scale cost aggregation for stereo matching. IEEE Transactions on Circuits and Systems for Video Technology, 2017, 27(5): 965–976. [doi: [10.1109/TCSVT.2015.2513663](https://doi.org/10.1109/TCSVT.2015.2513663)]
- 11 Hirschmüller H, Innocent PR, Garibaldi J. Real-time correlation-based stereo vision with reduced border errors. International Journal of Computer Vision, 2002, 47(1–3): 229–246.
- 12 Scharstein D, Szeliski R, Hirschmüller H. The middlebury stereo vision page. <http://vision.middlebury.edu/stereo/>.