

基于改进多权值滑动窗口的立体匹配方法的实现及应用

杜娟 沈思昀

(华南理工大学 广州 510641)

摘要 立体视觉的核心问题是通过立体匹配算法获得准确的视差值,然而大多数现有的立体匹配算法无法在低纹理或接近无纹理的区域中获得准确的视差值。为了解决低纹理区域匹配精度相对偏低、高精度半全局匹配算法运算量较大的问题,文中提出了一种基于自适应滑动窗口的立体匹配算法。首先通过 AD-Census 变换计算匹配代价,然后针对不同区域调节聚合窗口形状及像素点权重,最后结合符合人体视觉特征的多尺度代价聚合框架获得聚合匹配代价,采用赢者通吃策略获取最终的稠密视差图。实验结果证明,该算法在低纹理区域的误匹配率相比较传统方案的下降范围为 5.8%~21.68%,运算时间较半全局算法更短。

关键词 立体匹配,AD-Census 变换,高斯金字塔,自适应滑动窗口

中图分类号 TP391 文献标识码 A

Implementation and Application of Stereo Matching Method Based on Improved Multi-weight Sliding Window

DU Juan SHEN Si-yun

(South China University of Technology, Guangzhou 510641, China)

Abstract The key problem of stereo vision is to obtain accurate disparity values through stereo matching algorithms. However, most existing stereo matching algorithms are unable to obtain accurate and correct disparities in low-texture regions. In this paper, in order to solve the problems of low matching accuracy of low texture regions and large computational complexity of high-precision semi-global matching algorithm, a stereo matching algorithm based on adaptive sliding window was proposed. The cost volume is calculated by AD-Census transform firstly. The shape of the aggregate window and the weight of the pixels are adjusted for different regions. The cross-scale cost aggregation framework conforming to the human visual feature is used to obtain the aggregate cost volume. Finally, the “winner take all strategy” is used to obtain the final disparity maps. Experiments show that the mismatch rate of the algorithm in low-texture regions decrease from 5.8% to 21.68%, which is lower than that of the traditional scheme, and the computation time is shorter than the semi-global algorithm.

Keywords Stereo matching, AD-Census transform, Gaussian pyramid, Adaptive sliding window

1 引言

目前的立体匹配算法主要可分为全局匹配算法^[1-4]和局部匹配算法^[5-7]两类。局部匹配算法主要利用像素点及其周围的局部信息进行相应计算,虽然匹配正确率较低,但其计算复杂度低、计算量小、参数设置简单、执行效率高、实时性好。针对代价计算步骤,传统的基于亮度相似性约束的代价函数主要包括 AD(差值绝对值)、SAD(差值绝对值和函数)、SD(差值平方)、SSD(差值平方和函数),其函数构造简单,匹配精度较高,但对噪声敏感,鲁棒性较差。在此基础上,Zabih 等^[8]提出了 Census 变换法,能够较好地检测到图像中的边缘和角点等局部结构特征,且对光照变化较为鲁棒,但未考虑相似性约束,因此往往需要配合其他算法计算代价值来提高匹配精度;与其类似地,Shen^[9]提出采用秩变换(Rank Transform)计算代价值,以窗口内灰度值小于中心像素灰度值的像素个数来代替中心像素的灰度值;Hirschm^[3]和 Kim 等^[10]利用互信息(Mutual Information)作为匹配代价,但其运算包含

了对数与积分过程,运算复杂度较高。针对代价聚合步骤,最简单的方法是采用固定窗口(Fixed Window)作为支撑窗口进行代价聚合,但是在低纹理区域,往往需要设置较大的窗口,以囊括更多的像素值信息,而在复杂图像区域,则需要小窗口,以避免包含视差不连续的像素点,因此该方法的实际效果比较糟糕,但是其是许多代价聚合方法的基础,可以通过调整聚合窗口中心像素点位置^[11]、窗口大小及形状^[12]、窗口内像素点权重^[13]这几种思路来进行改进。Yoon 等^[13]提出了一种自适应支撑窗口(ASW)算法,根据指定窗口内的像素点与待匹配点间的色彩差值和几何距离来确定窗口内各像素点的权重,匹配结果精度较高,但由于需要对每一像素点进行权值计算,因此运行开销大。Zhang 等^[14]提出基于十字交叉聚合窗口的匹配算法,聚合窗口由多条相邻的水平分割线或者垂直分割线组成,通过阈值设定限制臂长的大小,来得到最佳的自适应窗口,但该方法对窗口内的像素点采用统一的权重,在斜面区域表现一般。由于支撑窗口法容易陷入局部最优解,Yang^[15]摒弃了该思想,提出了基于最小生成树(Minimum

Spanning Tree)的半全局算法,该方法充分利用了MST的性质,将全局像素点的比较转换成节点的比较,既充分考虑了全局像素点的影响,又大大缩短了运算时间。但是纹理丰富区域会导致最小生成树的构造出现错误,进而导致视差值不准确。在此算法的基础上,Mei等^[16]提前对图像进行分割,对各区域生成子树,最后利用贪心算法进行合并。该方法虽然在结果上略优于前者,但也增加了运行时间,且对整幅图像采用同一条件进行判断分割,分割效果一般。

为了解决低纹理区域匹配精度相对偏低、一般自适应支撑窗口法运算量较大的问题,本文提出了一种改进的多权重的滑动窗口法,在对图像区域进行划分的基础上,针对低纹理区域,选取较大窗口,针对复杂纹理区域,依据窗口内各像素点与中心像素的相似性进行权重分配,降低视差不连续点的影响,通过构建高斯金字塔,对匹配代价进行多尺度下的聚合,最终通过赢者通吃策略来获得精确的稠密视差图。该方案相对于ASW算法避免了对每一像素点进行权值分配,减少了运算量,相较于传统的基于亮度相似性约束的匹配算法,本方案在低纹理区域的误匹配率明显降低。

2 相关工作

2.1 AD-Census算法

基于颜色特征的绝对差(AD)对光照这类外界干扰因素比较敏感,但对于重复纹理区域的代价值计算不会受到影响,计算过程如式(1)所示:

$$C_{ad}(p, d) = \frac{1}{3} \sum_{i=R, G, B} |I_p(x, y) - I_p'(x-d, y)| \quad (1)$$

其中, $I_p(x, y)$ 和 $I_p'(x, y)$ 分别为参考图像 I 与匹配图像 I' 在点 $p = (x, y)$ 处的像素灰度值,求出的 $C_{ad}(p, d)$ 为参考图像 I 上点 $p = (x, y)$ 在视差 d 下的绝对差匹配代价。

Census变换则是对图像局部区域进行编码,在给定的窗口内比较中心像素与周围邻域像素之间的大小关系,灰度值小于中心点即记为1,灰度值大于中心点则记为0,每个像素都对对应一个二值编码序列,最后通过海明(Hamming)距离来表示两个像素的相似程度,因此对亮度变化和图像噪声等干扰因素具有较好的鲁棒性,然而在重复区域和相似纹理区域具有匹配模糊性,易导致误匹配,其计算如式(2)、式(4)所示,式(2)中的函数即比较两像素点的灰度大小,其含义如式(3)所示。

$$S(x, y) := \bigotimes_{i=-n'}^{n'} \bigotimes_{j=-m'}^{m'} \zeta(I(x, y), I(x+i, y+j)) \quad (2)$$

$$\zeta(a, b) = \begin{cases} 1, & \text{if } a > b \\ 0, & \text{if } a \leq b \end{cases} \quad (3)$$

$$C_{census}(p, d) := \text{Hamming}(S(x, y), S'(x-d, y)) \quad (4)$$

其中, n 和 m 为选取的邻域窗口的大小,且均为奇数; $S(x, y)$ 表示中心像素点 $p = (x, y)$ 的Census变换值; S 和 S' 分别为参考图像 I 与匹配图像 I' 上像素点的变换值; Hamming距离为两个比特串的对位不相同的数量; $C_{census}(p, d)$ 表示视差 d 下两比特串对位相异的个数,即点 $p = (x, y)$ 在视差 d 下的Census匹配代价。

基于两种匹配代价算法的优缺点,一个朴素的思想是尝试将二者进行线性融合^[17]。由于两种算法获取的初始代价值的范围相差较大,需要先对其进行归一化处理,然后求和获得融合的初始匹配代价值,其具体计算式如式(5)所示:

$$C(p, d) = 2 - \exp\left(-\frac{C_{census}(p, d)}{\lambda_{census}}\right) - \exp\left(-\frac{C_{ad}(p, d)}{\lambda_{ad}}\right) \quad (5)$$

其中, $C(p, d)$ 表示参考图像 I 上点 $p = (x, y)$ 在视差 d 下的总匹配代价,融合算法的处理效果图如图1所示。

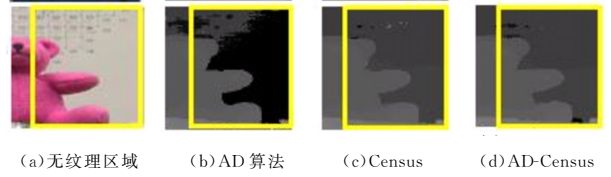


图1 不同匹配代价算法的处理效果图

2.2 高斯金字塔

图像金字塔^[18]是图像尺度表达的一种,是通过对原始图像进行多尺度像素采样的方式,生成 N 个不同分辨率的图像,以多分辨率来解释图像。传统意义上,一幅图像的金字塔是一系列以金字塔形状排列的分辨率逐步降低的来源于同一张原始图的图像集合,如图2所示。传统金字塔主要通过低通滤波器平滑图像和对平滑图像进行抽样(采样)两步骤获取。其层数 n 根据图像的原始大小和塔顶图像的大小共同决定。

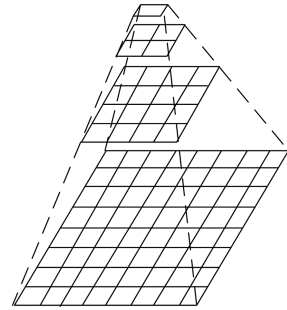


图2 传统图像金字塔

为了让尺度体现其连续性,高斯金字塔在简单降采样的基础上加上了高斯滤波。如图3所示,将图像金字塔每层的一张图像使用不同参数做高斯模糊,使得金字塔的每层含有多张高斯模糊图像,将金字塔每层多张图像合称为一组(Octave),组数和金字塔层数相等,每组含有多张(Interval)图像,在降采样时,高斯金字塔上一组图像的初始图像(底层图像)由前一组图像的倒数第三张图像隔点采样获得。

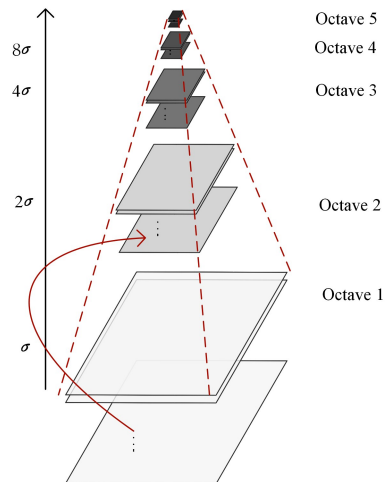


图3 高斯金字塔

2.3 跨尺度的代价聚合框架

人类从多个尺度处理立体对应关系,在人类立体视觉系统的对应搜索中交互地处理粗略和精细尺度的信息。从这种生物灵感来看,最终的匹配代价值是在这个框架中跨多个尺度进行汇总得出的。受人类立体视觉系统这一特点启发,采用多尺度代价聚合框架^[19]可以显著提高大多数传统代价聚合方法的性能,并获得更好的视差效果。

代价计算步骤如式(6)表示:

$$f: \mathbb{R}^{W \times H} \times \mathbb{R}^{W \times H} \rightarrow \mathbb{R}^{W \times H \times L} \quad (6)$$

其中, W 和 H 是输入图像的宽度和高度, L 表示视差水平的数量。

对于双目图像对 I 和 $I' \in \mathbb{R}^{W \times H}$,其代价值 $C \in \mathbb{R}^{W \times H \times L}$ 表示在所有可能的视差范围下每个像素的匹配成本,可以通过应用成本计算式: $C = f(I, I')$,即式(5)得到,但此时求得的成本量含有明显噪声,会影响后续代价聚合步骤结果的准确性,因此需对其进行去噪处理,从而得到去噪后的成本量 \tilde{C} ,其计算式如式(7)和式(8)所示:

$$\tilde{C}(p, d) = \underset{z}{\operatorname{argmin}} \sum_{Z_{p,j} \in N_p} K(p, q) \| z - C(q, d) \|^2 \quad (7)$$

$$Z_p = \sum_{q \in N_p} K(p, q) \quad (8)$$

其中, N_p 为点 p 的一邻域范围, $K(p, q)$ 是相似性内核,其度量的是像素点 p 和 q 之间的相似性,在本文算法中以 AD-Census 匹配代价衡量其相似性, Z_p 是归一化常数。

其次,将上标 s 添加到 \tilde{C} ,表示图像对在不同尺度下的代价值,其中 $s \in \{0, 1, \dots, S\}$ 为尺度参数。因此,式(7)和式(8)的多尺度版本可以写成式(9)和式(10):

$$\tilde{v} = \underset{z}{\operatorname{argmin}} \sum_{s=0}^S \frac{1}{Z_p^s} \sum_{Z_{p^s, q^s} \in N_{p^s}} K(p^s, q^s) \| z - C^s(q^s, d^s) \|^2 \quad (9)$$

$$Z_{p^s} = \sum_{q^s \in N_{p^s}} K(p^s, q^s) \quad (10)$$

其中, N_{p^s} 是归一化常数, $\{p^s\}_{s=0}^S$ 和 $\{l^s\}_{s=0}^S$ 为图4所示的每个尺度的对应变量序列, N_{p^s} 是第 s 个尺度下一组相邻的像素集。

跨尺度框架如图4所示,图中实线箭头表示尺度内一致性(常用于传统成本聚合方法),虚线箭头表示尺度间一致性,通过在不同尺度下执行的代价聚合的运算结果来进行自适应组合以获得最终匹配代价值。

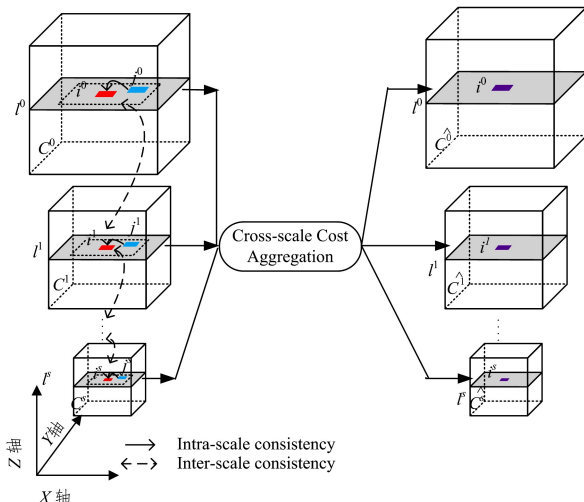


图4 跨尺度框架示意图

3 跨尺度下基于多权值滑动窗口的立体匹配算法

3.1 算法流程

3.1.1 区域分割

对最细尺度图像利用canny算子进行边缘检测,将具有较多边缘的复杂纹理区域以及无明显边缘的低纹理区域进行分割和标记。

使用如图5所示的边缘检测算子来获得检测图像中水平 G_x 和垂直 G_y 方向的一阶导数值,从而利用式(11)和式(12)确定像素点的梯度 G 和方向 θ ,确定图像的水平、垂直和对角边缘。

-1	0	1
-2	0	2
-1	0	1

(a) x方向算子 G_x

1	2	1
0	0	0
-1	-2	-1

(b) y方向算子 G_y

图5 边缘检测算子

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \quad (11)$$

$$\theta(x, y) = \arctan(G_x(x, y)/G_y(x, y)) \quad (12)$$

将当前像素的梯度强度与沿正负梯度方向上的两个像素进行比较(在跨越梯度方向的两个相邻像素之间使用线性插值得到要比较的像素梯度),如果当前像素的梯度强度与另外两个像素相比最大,则该像素点保留为边缘点,否则该像素点将被抑制。如果边缘像素的梯度值高于高阈值,则将其标记为强边缘像素;如果边缘像素的梯度值小于高阈值并且大于低阈值,则将其标记为弱边缘像素;如果边缘像素的梯度值小于低阈值,则进行抑制,不认定为边缘区域。最后查看弱边缘像素及其8邻域像素点,只要其中一个为强边缘像素,则将弱边缘区域保留,以此得到图像的边缘以及区域信息。

3.1.2 代价计算及代价聚合

对图像进行高斯滤波及降采样处理,获得图像的高斯金字塔,对各尺度图像由式(5)确定初始匹配代价值,利用式(13)进行第 s 层图像上的代价聚合,最后进行跨尺度的代价聚合。

$$C_s(p, d) = \sum_{n=1}^N \beta_n \sum_{p \in W_n(p)} C(p, d) \quad (13)$$

其中, W_n 为不同的窗口区域, β 为相应区域的权重。

3.1.3 视差图生成

对于待匹配图像 I 上像素点 p 的视差,本文使用“赢者通吃”(Winner-take-all, WTA)的策略,采用式(14)获得最终视差,其中 D 为最大视差搜索范围。

$$D(p) = \underset{d \in D}{\operatorname{argmin}} C(p, d) \quad (14)$$

本文算法的优点在于,避免了对每一像素点进行权重分配,减少了运算量,相较于传统的基于亮度相似性约束的匹配算法,加入了非参数变换求得匹配代价,同时针对不同区域赋予不同权重,在低纹理区域的误匹配率明显降低。

3.2 进行权重分配的自适应窗口生成方法

滑动窗口法的中心思想是取图像对的像素点的周围 $N \times N$ 大小的邻域,分别以这个领域中的所有点为中心,使用窗口法得到 $N \times N$ 个匹配代价中的最优值作为最终的匹配代价。本文主要针对图像的不同区域调整滑动窗口的大小来适应图像不同区域的特征,并调整滑动窗口中每个区域的权重以增加相关性较强区域的权重占比,提升结果的准确性。

3.2.1 低纹理区域的聚合窗口生成及权重系数设定

针对低纹理区域设置较大的窗口,以获取足够的像素点信息,由于不存在视差不连续的情况,可以基于距离中心像素的距离设定权值来避免对大窗口内的每一像素点计算权重,以大量减少运算量。式(13)中的 W_n 即为图6所示的窗口,确定权重 β_n 时令 $\beta_1 > \beta_2 > \beta_3$ (本文中取值为 $\beta_1 = 0.6, \beta_2 = 0.3, \beta_3 = 0.1$)。

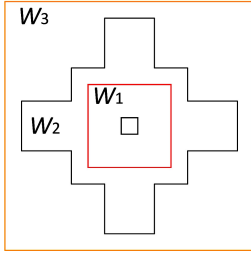


图6 低纹理区域窗口

3.2.2 复杂纹理区域的聚合窗口生成及权重系数设定

由于复杂区域内的中心像素点大概率是视差不连续点,若继续采用传统矩形窗口会不可避免地引入不同视差下的像素点,降低匹配精度,因此本文针对8个梯度方向设计了如图7所示的8个活动窗口,使得在任意位置的中心像素点均能有相应的匹配窗口以充分囊括有效区域。由于在局部区域内,颜色越相似的两个像素越有可能处于同一视差下,因此本文选择窗口内颜色的均值与中心像素点 p 的颜色最为相近的3个窗口作为有效窗口,即式(13)中的 W_n ,在进行代价聚合时在有效窗口上进行加权聚合。

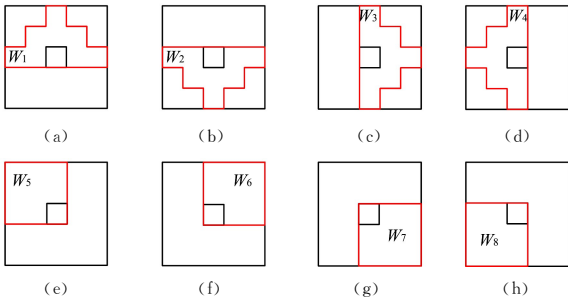


图7 复杂纹理区域聚合窗口

8个候选窗口内的颜色均值采用如下方案进行计算:针对图7(a)~图7(d)4个窗口,将窗口划分为3块进行均值计算,如式(15)~式(18)所示:

$$\text{mean}_{W_1}^i(I) = \frac{5}{9} * \text{mean}I_{1 \times 5}^i(x, y) + \frac{3}{9} * \text{mean}I_{1 \times 3}^i(x, y-1) + \frac{1}{9} * I^i(x, y-2) \quad (15)$$

$$\text{mean}_{W_2}^i(I) = \frac{5}{9} * \text{mean}I_{1 \times 5}^i(x, y) + \frac{3}{9} * \text{mean}I_{1 \times 3}^i(x, y+1) + \frac{1}{9} * I^i(x, y+2) \quad (16)$$

$$\text{mean}_{W_3}^i(I) = \frac{5}{9} * \text{mean}I_{1 \times 5}^i(x, y) + \frac{3}{9} * \text{mean}I_{1 \times 3}^i(x+1, y) + \frac{1}{9} * I^i(x+2, y) \quad (17)$$

$$\text{mean}_{W_4}^i(I) = \frac{5}{9} * \text{mean}I_{1 \times 5}^i(x, y) + \frac{3}{9} * \text{mean}I_{1 \times 3}^i(x-1, y) + \frac{1}{9} * I^i(x-2, y) \quad (18)$$

其中, i 代表 R, G, B 3个颜色通道, $\text{mean}I_{1 \times n}^i(x, y)$ 表示采用

$1 * n$ 大小的过滤窗进行均值过滤后的值。

针对图7(e)~图7(h)4个窗口,其均值计算式如式(19)~式(22)所示:

$$\text{mean}_{W_5}^i(I) = \text{mean}I_{3 \times 3}^i(x-1, y-1) \quad (19)$$

$$\text{mean}_{W_6}^i(I) = \text{mean}I_{3 \times 3}^i(x+1, y-1) \quad (20)$$

$$\text{mean}_{W_7}^i(I) = \text{mean}I_{3 \times 3}^i(x+1, y+1) \quad (21)$$

$$\text{mean}_{W_8}^i(I) = \text{mean}I_{3 \times 3}^i(x-1, y+1) \quad (22)$$

对得到的3个有效窗口 W_{n1}, W_{n2}, W_{n3} ($\text{mean}_{W_{n1}} > \text{mean}_{W_{n2}} > \text{mean}_{W_{n3}}$), 将其权重 $\beta_1, \beta_2, \beta_3$ 按式(23)进行设置:

$$\beta_i = \frac{1}{2} * \left(1 - \frac{\text{mean}_{W_{ni}}}{\text{mean}_{W_{n1}} + \text{mean}_{W_{n2}} + \text{mean}_{W_{n3}}} \right) \quad (23)$$

4 实验结果及分析

本次实验采用 Middlebury 数据集中的 Teddy 及 Cones 图像对,测试了该方法的性能并将结果与使用其他代价聚合方案的立体匹配算法进行比较,包括盒式滤波器方法、双边滤波器方法和分段树方法。为了避免后续步骤带来的对匹配精度测量的干扰,不对获得的视差图进行视差细化处理,仅通过赢者通吃策略生成最终的视差图。实验结果如图8~图10所示。

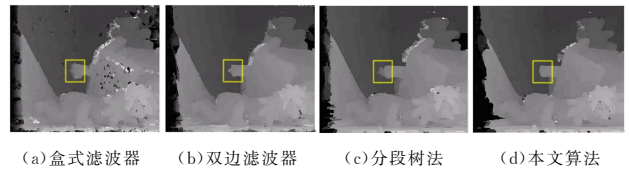


图8 各匹配算法获得的视差图像(电子版为彩色)

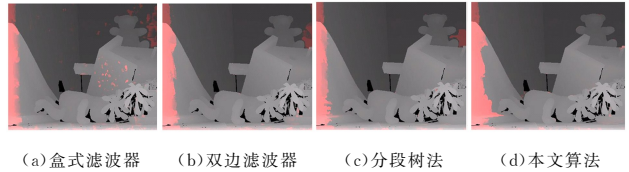


图9 各匹配算法获得的视差与真实视差对比图



图10 低纹理区域视差对比图(电子版为彩色)

4.1 匹配误差数据统计

表1、表2分别列出了采用 AD, Census 及 AD-Census 算法获得的视差图的质量定量评估结果,其中表1所列误匹配率数据均为有效区域的误匹配率,表2所列低纹理区域的误匹配率。表3、表4列出了在采用 AD-Census 进行代价计算后,分别采用不同代价聚合算法获取到的视差图的评估结果。其中,“avg error”表示整幅图像中误匹配像素点的百分比,“error”表示图像中无法匹配区域以外的误匹配率,“text1 error”为遮挡及无纹理区域的误匹配率。

表1 采用不同代价计算算法的图像有效区域误匹配率

Teddy	盒式滤波器	双边滤波器	分段树	本文方案
AD	7.87	7.39	7.13	7.15
Census	6.82	6.02	5.91	5.90
AD-Census	6.53	5.68	5.52	5.36

表 2 采用不同代价计算算法的低纹理区域误匹配率

Teddy	盒式滤波器	双边滤波器	分段树	本文方案
AD	12.36	11.52	10.71	10.96
Census	10.79	10.44	9.21	9.03
AD-Census	9.87	8.21	7.95	7.73

表 3 采用不同代价聚合算法的 Teddy 图像误匹配率

Teddy	盒式滤波器	双边滤波器	分段树	本文方案
Avg error	18.13	15.74	15.57	15.73
error	6.53	5.68	5.52	5.36
Textl error	9.87	8.21	7.95	7.73

表 4 采用不同代价聚合算法的 Cones 图像误匹配率

Cones	盒式滤波器	双边滤波器	分段树	本文方案
Avg error	15.83	15.55	14.97	14.49
error	6.03	5.42	5.10	5.07

由表 1、表 2 中的数据对比可以看出,采用 AD-Census 算法进行匹配代价计算可以有效提高图像匹配精度,全图像有效区域的匹配误差降低了 4%~9%,低纹理区域的匹配精度提升更为明显。表 3、表 4 中的数据则显示,在均未采用左右一致性检验的情况下,采用盒式滤波器法进行代价聚合的误差在全图及低纹理区域普遍较大,采用分段树法的匹配精度均相对较高,但这一方法属于半全局算法,相较于其他算法计算量明显加大,在牺牲了一定实时性的基础上获得了更好的匹配效果。相较于盒式滤波器法及双边滤波器法,采用多权重滑动窗口的全图像有效区域误匹配率下降了 5.63%~17.92%,低纹理区域误匹配率下降了 5.85%~21.68%,匹配精度有一定提升。从图 8 的黄色框及图 10 的红色框所限定的区域可以明显看出,本文方案在视差不连续区域及低纹理区域的表现均较为良好,对不同区域调节聚合窗口形状及像素点的权重可以有效地兼顾算法在纹理复杂区域及低纹理区域的表现。

4.2 运行时间数据统计

本次实验环境为 Intel 酷睿 i7 5700HQ CPU,2.7 GHz 主频,运行环境为 Visual Studio 2015,Open CV 3.3.0。表 5 列出了不同代价计算与不同代价聚合方案相结合来获取最终视差图的运行时间。

表 5 采用不同立体匹配算法的运行时间

Teddy	盒式滤波器	双边滤波器	分段树	本文方案
Census	52.07'	54'21.10"	59.53"	52.72'
AD-Census	59.07'	59'25.47"	1'6.85"	57.86'

从表 5 中可以看出,采用双边滤波器方法进行代价聚合时,由于其不是线性复杂度,当图像尺寸较大、选取的窗口增大时运算量大幅增加,耗时长;相较于半全局的分段树法,本文方案在运算效率上有一定提升;而盒式滤波器的算法由于其复杂度为 $O(1)$,因此针对大尺寸图像的处理也有高效率,本文方案在实时性上并无优势,但盒式滤波器无法通过多尺度框架进行视差的优化,因此本文方案在精度上相对其有一定优势。

结束语 本文提出的改进的多权重的滑动窗口法,通过对图像进行区域划分,针对纹理复杂区域及低纹理区域采用不同的策略,修改聚合窗口的形状及窗口内各像素点的权重,最终获得在各区域均有较高匹配精度的视差图。与匹配精度高的全局、半全局匹配算法相比减少了运算量,相较于传统的基于亮度相似性约束的匹配算法有效地降低了误匹配率。

参考文献

- [1] YANG Q, WANG L, YANG R, et al. Stereo Matching with Color-Weighted Correlation, Hierarchical Belief Propagation and Occlusion Handling[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. IEEE Computer Society, 2006.
- [2] HONG L, CHEN G. Segment-Based Stereo Matching Using Graph Cuts[C]// null. IEEE Computer Society, 2004.
- [3] HIRSCHM H. Stereo Processing by Semi-global Matching and Mutual Information[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 30(2): 328-341.
- [4] ŽBONTAR, JURE, LECUN Y. Computing the Stereo Matching Cost with a Convolutional Neural Network[J]. Computer Vision and Pattern Recognition, 2014.
- [5] YANG Q. Stereo Matching Using Tree Filtering [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2015.
- [6] HE K, JIAN S, TANG X. Guided Image Filtering[M]// Computer Vision- ECCV 2010. 2010.
- [7] KANADE T, OKUTOMI M. A stereo matching algorithm with an adaptive window: theory and experiment[C]// IEEE International Conference on Robotics & Automation. IEEE, 2002.
- [8] ZABIH R, WOODFILL J. Non-parametric Local Transforms for Computing Visual Correspondence [J]. Proc. third European Conf. computer Vision, 1994: 151-158.
- [9] SHEN Y. Efficient normalized cross correlation calculation method for stereo vision based robot navigation[J]. 中国计算机科学前沿: 英文版, 2011, 5(2): 227-235.
- [10] KIM J, KOLMOGORO V, ZABI H. Visual correspondence using energy minimization and mutual information[C]// Proceedings Ninth IEEE International Conference on Computer Vision. IEEE, 2008.
- [11] SCHARSTEIN D, SZELISKI R. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms[J]. International Journal of Computer Vision, 2002, 47(1/3): 7-42.
- [12] VEKSLER O. Fast variable window for stereo correspondence using integral images[C]// 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. IEEE, 2003.
- [13] YOON K J, KWEON I S. Adaptive Support-Weight Approach for Correspondence Search[J]. IEEE Trans Pattern Anal Mach Intell, 2006.
- [14] ZHANG K, LU J, LAFUIT G. Cross-Based Local Stereo Matching Using Orthogonal Integral Images[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2009, 19(7): 1073-1079.
- [15] YANG Q. A non-local cost aggregation method for stereo matching[C]// 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2012.
- [16] MEI X, SUN X, DONG W, et al. Segment-Tree Based Cost Aggregation for Stereo Matching[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2013.
- [17] MEI X, SUN X, ZHOU M, et al. On building an accurate stereo matching system on graphics hardware[C]// IEEE International Conference on Computer Vision Workshops, 2012.
- [18] LOWE D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110.
- [19] ZHANG K, FANG Y, MIN D, et al. Cross-Scale Cost Aggregation for Stereo Matching[C]// Computer Vision & Pattern Recognition. IEEE, 2014.