

8-14-2020

Stereo Matching based on Pyramid transform Cross-Scale Cost Aggregation

Yao Li

1. Key Laboratory of Computer Network and Information Integration (Southeast University), Nanjing 211189, China;;2. State Key Laboratory for Novel Software Technology(Nanjing University), Nanjing 210032, China;;3. School of Computer Science and Engineering, Southeast University, Nanjing 211189, China;

Zhukui Liu

3. School of Computer Science and Engineering, Southeast University, Nanjing 211189, China;

Bingfeng Wang

3. School of Computer Science and Engineering, Southeast University, Nanjing 211189, China;

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research](#), [Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Stereo Matching based on Pyramid transform Cross-Scale Cost Aggregation

Abstract

Abstract: Human visual system processes the received visual signals on different scales, however, the traditional stereo matching algorithms get the disparity map from the original image in the biggest scale, which will lead to high stereo matching error rate in the area of low-texture, texture-less area. Simulation of the human visual system in stereo matching on multiple scales can reduce error rate. *A stereo matching method based on the Gaussian pyramid cross-scale transform was improved, by adding the Laplace Pyramid Transform to the original cross-scale framework and adding the weighted joint bilateral filter in the disparity refinement stage.* The new cross-scale framework can get a better disparity map than the original method.

Keywords

stereo matching, cross-scale, pyramid transform, weighted joint bilateral filter, disparity refinement

Recommended Citation

Yao Li, Liu Zhukui, Wang Bingfeng. Stereo Matching based on Pyramid transform Cross-Scale Cost Aggregation[J]. Journal of System Simulation, 2016, 28(9): 2227-2234.

基于金字塔变换跨尺度代价聚合的立体匹配

姚莉^{1,2,3}, 刘助奎³, 王秉凤³(1. 计算机网络和信息集成教育部重点实验室(东南大学), 南京 211189; 2. 计算机软件新技术国家重点实验室(南京大学), 南京 210023;
3. 东南大学计算机科学与工程学院, 南京 211189)

摘要: 人眼视觉系统在不同尺度上处理接收到的视觉信号, 而传统的立体匹配算法都在原立体匹配图像对最大尺度下进行立体匹配, 在低纹理区域、无纹理区域立体匹配错误率高。模拟人眼视觉系统在多个尺度上进行立体匹配, 再融合不同尺度下的匹配结果能减少错误率。通过改进一种基于高斯金字塔变换跨尺度代价聚合的立体匹配方法, 在原框架上, 在代价聚合阶段加入拉普拉斯金字塔变换, 在视差精细化阶段加入基于加权联合双边滤波的视差精细化方法, 使新框架能获得比原论文更好的视差图。

关键词: 立体匹配; 跨尺度; 金字塔变换; 加权联合双边滤波; 视差精细化

中图分类号: TP391.4

文献标识码: A

文章编号: 1004-731X (2016) 09-2227-08

Stereo Matching based on Pyramid transform Cross-Scale Cost Aggregation

Yao Li^{1,2,3}, Liu Zhukui³, Wang Bingfeng³(1. Key Laboratory of Computer Network and Information Integration (Southeast University), Nanjing 211189, China;
2. State Key Laboratory for Novel Software Technology(Nanjing University), Nanjing 210032, China;
3. School of Computer Science and Engineering, Southeast University, Nanjing 211189, China)

Abstract: Human visual system processes the received visual signals on different scales, however, the traditional stereo matching algorithms get the disparity map from the original image in the biggest scale, which will lead to high stereo matching error rate in the area of low-texture, texture-less area. Simulation of the human visual system in stereo matching on multiple scales can reduce error rate. A stereo matching method based on the Gaussian pyramid cross-scale transform was improved, by adding the Laplace Pyramid Transform to the original cross-scale framework and adding the weighted joint bilateral filter in the disparity refinement stage. The new cross-scale framework can get a better disparity map than the original method.

Keywords: stereo matching; cross-scale; pyramid transform; weighted joint bilateral filter; disparity refinement

引言

立体匹配是通过查找不同视角拍摄的两幅或者多幅同一场景不同图像间的匹配像素点, 然后将匹配像素点之间的 2D 位移差(也叫视差 disparity)转

换为 3D 深度, 从而估算出场景的一个 3D 模型^[1]。

立体匹配广泛应用于虚拟现实、机器人导航、3D 场景绘制与重建、大型机械姿态感知等领域^[2]。

Scharstein 等人^[3]建立了立体匹配的评价体系, 提供了相应的测试数据集; 指出立体匹配通常都要进行如下 4 个步骤中的部分操作: 匹配代价(cost)计算、代价(支持)聚合(aggregation)、视差计算和优化、视差精细化(refinement)。

1) 在代价计算阶段, 通过计算同一场景不同



收稿日期: 2016-05-26 修回日期: 2016-07-14;
基金项目: 国家高技术研究发展计划(863 计划) (2015 AA015904), CCF-腾讯犀牛鸟基金(RAGR20150120), NSFC-广东联合基金(第二期) 超级计算科学应用研究专项资助和国家超级计算广州中心支持;
作者简介: 姚莉(1977-), 女, 湖北, 博士, 副教授, 研究方向为计算机图形学与视觉。

<http://www.china-simulation.com>

• 2227 •

视角两幅图像最大视差 $d_{\max}$ ($d_{\max}$ 由拍摄两幅图像相机的间距即基线长度决定) 范围内, 每个视差下两幅图像对应像素点之间的匹配代价, 匹配代价用来衡量像素点之间相似性的大小, 通过匹配代价计算能得 $d_{\max}$ 个与原图像分辨率一样大的匹配代价图所组成的匹配代价卷(cost volume);

2) 由于匹配代价计算得到的匹配代价图包含噪声, 在代价聚合阶段通过像素点为中心的支持窗口内其他像素点匹配代价来修正当前像素点的匹配代价, 隐含假设了代价聚合具有平滑性;

3) 在视差计算和优化阶段, 使用 WTA (Winner Take All) 算法, 从匹配代价卷的 $d_{\max}$ 个匹配代价图中, 取每个像素点的最小匹配代价所对应的视差作为该像素点的视差, 此时匹配代价最小, 两幅图像对应视差下的像素点对最相似最匹配。

4) 在视差精细化阶段则采用相关的算法对前三步提取到的视差图进行修正, 以得到错误更少, 场景中物体边缘轮廓信息保持更好的视差图。

在 4 个操作步骤中, 代价聚合阶段对最终得到的视差图质量影响最大, 目前大部分的立体匹配研究都着重于代价聚合方面算法的改进与创新上。

大部分的代价聚合算法隐含假设了代价聚合具有平滑性, 可以看成是在匹配代价卷上进行滤波。早期简单高效的局部线性滤波器如盒滤波器(box filter)、高斯滤波器(Gaussian filter)被用来在代价聚合阶段对匹配代价卷的处理, 但它们是各向同性的滤波器, 也就是对图像的横向和纵向作用强度相同, 会导致最终得到的视差图中物体边缘轮廓部分出现模糊。之后具有保边性的双边滤波器(bilateral filter)^[4]以及引导滤波器(guided filter)^[5]被引入到立体匹配代价聚合阶段对匹配代价卷的滤波处理, 双边滤波器的计算量与其采用的核窗口大小有关, 核窗口越大则滤波器计算量也越大, 要取得好的滤波效果需要采用大的核窗口; 引导滤波器的计算量与核窗口大小无关, 计算量相比双边滤波器小, 目前已经有使用 GPU 加速版本的双边滤波器和引导滤波器。近年来将核

窗口扩展至整个图像的非局部方法如最小生成树(minimum spanning tree: MST)^[6]、分段树/区间树(segment tree: ST)^[7]被引入到立体匹配代价聚合阶段对匹配代价卷的处理, 它们能比局部的滤波器取得更好的视差图。

这些最新的立体匹配代价聚合处理算法推动了立体匹配的发展, 但是目前都存在一个问题: 它们都是在图像的最大分辨率下进行处理, 没有考虑人眼视觉系统是在不同尺度上处理接收到的视觉信号^[8], 模拟人眼视觉系统由粗到精(coarse to fine)的处理过程。

Zhang 等人在文献[9]中首次将高斯金字塔变换多尺度分解方法引入到立体匹配代价聚合阶段的处理, 搭建了一个跨尺度代价聚合框架, 有效提高了采用局部代价聚合方法如盒滤波器、高斯滤波器、双边滤波器、引导滤波器, 以及非局部代价聚合方法如最小生成树、分段树提取到的视差图质量。但他们的方法是采用高斯金字塔变换分解原图像后, 在各高斯金字塔分解层下进行匹配代价计算和代价聚合, 然后将各个高斯金字塔分解层下求得的匹配代价直接进行融合得到最终的匹配代价, 这样会造成在粗尺度下平滑过渡而给最终融合得到的匹配代价引入失真。

本论文受 Zhang 等人的启发, 借鉴 Qin 等人^[10-11]的能量函数设计思想, 在原论文基于高斯金字塔变换的框架上, 引入拉普拉斯金字塔变换, 在融合高斯金字塔各个分解层上的匹配代价之前进行拉普拉斯上采样再融合匹配代价, 有效地提高视差图的质量; 在视差精细化阶段引入基于加权联合双边滤波的视差精细化方法替换原论文中的加权中值滤波方法, 进一步提高最终视差图的质量。

1 基于金字塔变换跨尺度的立体匹配

本文算法流程如图 1 所示, 步骤如下:

1) 在立体匹配图像对 I 、 I' 上分别构建 $S+1$ 层高斯金字塔, 随着分辨率的降低, 高斯金字塔低

分辨率分解层上的最大视差也相应减小;

2) 在高斯金字塔各分解层下分别进行匹配代价计算和代价聚合操作得到各个尺度下的匹配代价卷, 并对低分辨率分解层上的匹配代价卷进行拉普拉斯上采样, 使其分辨率与原图像 I 、 I' 一致;

3) 融合各个尺度分辨率下的匹配代价卷, 得

到最终与原图像 I 、 I' 分辨率一致的匹配代价卷;

4) 在最终匹配代价卷上使用 WTA 算法计算视差, 得到初始视差图;

5) 对初始视差图使用基于加权联合双边滤波的视差精细化方法进行精细化, 得到最终的视差图。

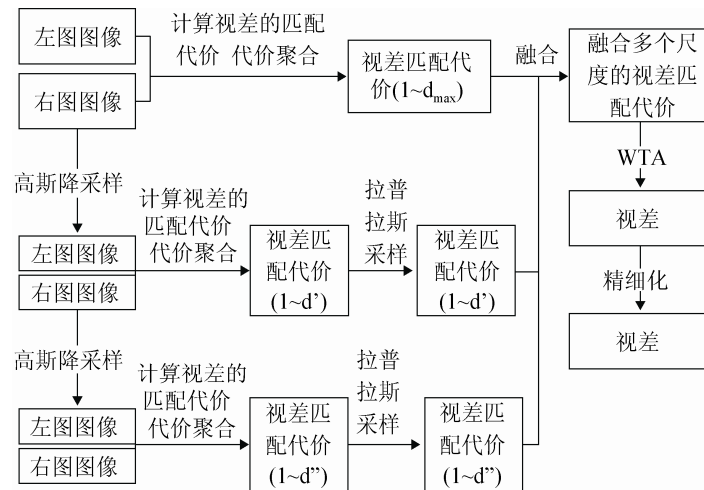


图 1 金字塔变换跨尺度代价聚合视差图提取流程

1.1 计算匹配代价

匹配代价是用来衡量不同视角拍摄的两幅或者多幅同一场景不同图像, 在不同视差下对应像素点对之间在的相似性。匹配代价计算使用 $f: R^{W \times H \times 3} \times R^{W \times H \times 3} \rightarrow R^{W \times H \times L}$ 表示, 其中 W 、 H 表示图像分辨率的宽和高, 3 代表像素 RGB 通道, L 代表最大视差 $d_{\max}$ 。不同视角拍摄的两幅同一场景不同立体图像对 I 、 I' 使用式(1)来表示在视差从到的匹配代价卷。

$$C = f(I, I') \quad (1)$$

对于 x_i, y_i 坐标处的像素点 $i = f(x_i, y_i)$, 在视差 l 下的匹配代价为标量 $C(i, l)$ 。使用亮度加梯度的匹配代价为目前常用的匹配代价计算方法:

$$C(i, l) = (1 - \alpha) \cdot \min(\|I(i) - I'(i_l)\|^2, \tau_c) + \alpha \cdot \min(\|\nabla_x I(i) - \nabla_x I'(i_l)\|^2, \tau_g) \quad (2)$$

式中: $I(i)$ 表示图像 I 中像素点 i 的 RGB 颜色向量; ∇_x 表示 x 方向的亮度梯度; τ_c 、 τ_g 分别为亮度和 x 方向亮度梯度的截断参数; $i_l = (x_{i-l}, y_i)$ 为像素点

i 在视差 l 下在图像 I' 中的对应像素点。

利用式(2)计算 I 中每个像素点在视差 $l(1 \leq l \leq L)$ 下的匹配代价即可得到匹配代价图, 重复计算每个视差下的匹配代价图即可得 L 个匹配代价图所组成的匹配代价卷, 匹配代价卷可以看成是一个维度为 $W \times H \times L$ 的三维矩阵。

1.2 代价聚合

由计算匹配代价阶段获得的匹配代价卷包含噪声, Milanfar 在文献[12]指出, 代价聚合阶段对匹配代价卷的去噪声处理可以看成是对匹配代价卷的加权最小二乘(weighted least squares: WLS)优化问题。本文采用的 WLS 优化如式(3)所示:

$$\tilde{C}(i, l) = \operatorname{argmin}_z \frac{1}{Z_i} \sum_{j \in N_i} K(i, j) \|z - C(j, l)\|^2 \quad (3)$$

其中当代价聚合采用局部方法时 N_i 为以像素点 i 为中心核窗口内的其他像素点, 当采用非局部的方法时 N_i 为整个图像内的其他像素点; $K(i, j)$ 为用来衡量像素点 i, j 之间相似性的相似性核, 在使用

引导滤波器做相似核函数时采用像素点间像平面二维空间距离和颜色空间距离来衡量像素点 i, j 之间的相似性, 在使用最小生成树做相似核函数时则采用测地线距离衡量像素点 i, j 之间的相似性; $Z_i = \sum_{j \in N_i} K(i, j)$ 为归一化常数; z 变量为匹配代价迭代去噪声过程中的输出值; 式(4) $\tilde{C}(i, l)$ 为匹配代价迭代去噪声过程的最终输出值, 即最终去除噪声的匹配代价, 也就是说当 $z = \tilde{C}(i, l)$ 时为式(3)加权最小二乘优化问题的最优解。

$$\tilde{C}(i, l) = \frac{1}{Z_i} \sum_{j \in N_i} K(i, j) C(j, l) \quad (4)$$

1.3 跨尺度的代价聚合框架

当前的局部和非局部代价聚合算法都是在立体图像对 I, I' 原始分辨率下的匹配代价卷上进行代价聚合, 使用这些算法在高纹理区域都能取得质量好的视差值, 但是在低纹理区域或者无纹理区域则效果都不好, 考虑到人眼视觉系统是在不同尺度上处理接收到的视觉信号, 并且对细节信息非常敏感, 本文将原立体图像对 I, I' 分解到多个尺度上, 在各个尺度分解层下分别进行代价聚合处理再融合各个尺度上的代价聚合结果, 能获得更好的视差图。

金字塔变换是常用的跨尺度多分辨率分解方法。一幅图像的金子塔, 是该图像经过相应处理再降采样形成的一系列以金字塔形状排列, 分辨率逐步降低的图像集合, 金字塔顶部是待处理图像低分辨率表示, 其他金字塔层是在不同尺度下的高分辨率表示。图像的不同金字塔分解层, 能用来分析图像中不同大小的物体, 其中低分辨率图像可用来分析图像中大的物体, 高分辨率图像可用来分析图像中的小的物体以及细节信息。图像的重要特征按照不同的尺度, 不同的分辨率分解到不同的金字塔分解层上。按照金字塔构造原理的不同, 金字塔可以分为高斯金字塔、拉普拉斯金字塔、梯度金字塔、对比度金字塔等^[8]。

对于高斯金字塔, 第分解层图像与第分解层图像之间满足式(5):

$$G_l(i, j) = \sum_{m=-\frac{k-1}{2}}^{\frac{k-1}{2}} \sum_{n=-\frac{k-1}{2}}^{\frac{k-1}{2}} w(m, n) G_{l-1}(2i+m, 2j+n) \quad (5)$$

其中 $1 \leq i \leq R_{l-1}/2$, $1 \leq j \leq C_{l-1}/2$, R_{l-1} , C_{l-1} 为第 $l-1$ 分解层图像 G_{l-1} 的行数和列数; 第分解层图像 G_l 比第 $l-1$ 分解层图像 G_{l-1} 缩小了 4 倍; $w(m, n)$ 为大小为奇数 $k(k=3, 5, 7, \dots)$ 的生成核, 是一个窗函数, 可以看成是一个低通二维滤波器, 一个典型的 $5 \times 5 w(m, n)$ 窗口如式(6)所示

$$w = \frac{1}{2} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} \quad (6)$$

图像的拉普拉斯金字塔可以通过求解高斯金字塔中每两层图像之间的差异得到, 为求出高斯金字塔第 l 分解层图像 G_l 与第 $l-1$ 分解层图像 G_{l-1} 之间的差异, 需要先把低分辨率高斯金字塔分解层图像 G_l 进行扩充放大, 使它的分辨率和高分辨率分解层图像 G_{l-1} 一样, 定义扩大算子 Expand:

$$G_l^* = \text{Expand}(G_l) \quad (7)$$

通过扩大算子运算得到的 G_l^* 与 G_{l-1} 分辨率一样, 扩大算子的具体运算是通过对第 l 分解层图像 G_l 进行插值放大, 由式(8)实现:

$$G_l^*(i, j) = 4 \sum_{m=-\frac{k-1}{2}}^{\frac{k-1}{2}} \sum_{n=-\frac{k-1}{2}}^{\frac{k-1}{2}} w(m, n) G_l' \left(\frac{i-m}{2}, \frac{j-n}{2} \right) \quad (8)$$

其中 $1 \leq i \leq R_{l-1}$, $1 \leq j \leq C_{l-1}$, R_{l-1} , C_{l-1} 为第 $l-1$ 分解层图像 G_{l-1} 的行数和列数, G_l' 由式(9)决定:

$$G_l' \left(\frac{i-m}{2}, \frac{j-n}{2} \right) = \begin{cases} G_l \left(\frac{i-m}{2}, \frac{j-n}{2} \right), & \frac{i-m}{2}, \frac{j-n}{2} \text{ 为整数} \\ 0, & \text{其他} \end{cases} \quad (9)$$

完整的拉普拉斯金字塔定义为:

$$\begin{cases} LP_l = G_l - \text{Expand}(G_{l+1}), & 0 \leq l < N \\ LP_N = G_N, & l = N \end{cases} \quad (10)$$

将原立体图像对 I 、 I' 使用高斯金字塔变换分解到 $0, 1, \dots, S$ 高斯分解层; 分别在第 $0, 1, \dots, S$ 高斯分解层下, 分别计算视差从 1 到 L 、 1 到 $L/2$ 、 1 到 $L/2^2 \dots 1$ 到 $L/2^S$ 的匹配代价并独立进行代价聚合, 得到匹配代价卷 $C^0, C^1, C^2 \dots C^S$, 其中 C^0 是在最大分辨率下的匹配代价卷, C^S 是在最小分辨率下的代价卷; 之后对低分辨率 $1, 2, \dots, S$ 高斯分解层上的匹配代价卷 $C^1, C^2 \dots C^S$ 进行拉普拉斯上采样 $\text{Expand}^{(1)}(C^1)$, $\text{Expand}^{(2)}(C^2)$, $\dots, \text{Expand}^{(S)}(C^S)$, 使低分辨率高斯分解层上的匹配代价卷通过多次拉普拉斯上采样后分辨率和原图像 I 、 I' 以及 C^0 一样, $\text{Expand}^{(s)}$ 表示使用 s 次拉普拉斯上采样使低分辨率代价卷恢复到原始图像的分辨率。引入高斯金字塔变换^[9]的式(3)、(4)修正为式(11)、(12):

$$\tilde{\mathbf{v}} = \arg \min_{\{z^s\}_{s=0}^S} \sum_{s=0}^S \frac{1}{Z_{i^s}^s} \sum_{j^s \in N_{i^s}} K(i^s, j^s) \|z^s - \text{Expand}^{(s)}(C^s(j^s, l^s))\|^2 \quad (11)$$

$$\forall s, \tilde{C}^s(i^s, l^s) = \frac{1}{Z_{i^s}^s} \sum_{j^s \in N_{i^s}} K(i^s, j^s) \text{Expand}^{(s)}(C^s(j^s, l^s)) \quad (12)$$

其中 $Z_{i^s}^s = \sum_{j^s \in N_{i^s}} K(i^s, j^s)$ 是尺度 s 下的归一化常数; $\{i^s\}_{s=0}^S$ 、 $\{j^s\}_{s=0}^S$ 、 $\{l^s\}_{s=0}^S$ 为尺度 s 下对应的变量; $\{z^s\}_{s=0}^S$ 为代价聚合操作(迭代去噪声过程)的中间输出值; 式(13) $\tilde{\mathbf{v}}$ 表示在尺度 $0, 1, \dots, S$ 下分别独立执行代价聚合操作, 并对代价聚合结果进行拉普拉斯上采样得到最终的匹配代价向量。

$$\tilde{\mathbf{v}} = [\tilde{C}^0(i^0, l^0), \tilde{C}^1(i^1, l^1) \dots \tilde{C}^S(i^S, l^S)]^T \quad (13)$$

为了融合多个尺度的匹配代价, 引入不同尺度之间的正则化约束项^[9]到式(11), 保证相邻尺度同一像素匹配代价的一致性, 如式(14)所示:

$$\hat{\mathbf{v}} = \underset{\{z^s\}_{s=0}^S}{\text{argmin}} \sum_{s=0}^S \frac{1}{Z_{i^s}^s} \sum_{j^s \in N_{i^s}} K(i^s, j^s) \|z^s - \text{Expand}^{(s)}(C^s(j^s, l^s))\|^2 + \lambda \sum_{s=1}^S \|z^s - z^{s-1}\|^2 \quad (14)$$

其中 λ 为正则化常数因子, 正则化常数因子越大, 同一像素不同尺度之间的一致性约束越强,

会加强对低纹理区域的视差估计(低纹理区域视差难以估计, 需要加强约束才行), 但是副作用是使得其他区域的视差值估计不够精确; $\hat{\mathbf{v}}$ 表示加入不同尺度之间的正则化约束项后在尺度 $0, 1, 2, \dots, S$ 下经过代价聚合操作(迭代去噪声过程最后得到)的匹配代价组成的向量:

$$\hat{\mathbf{v}} = [\hat{C}^0(i^0, l^0), \hat{C}^1(i^1, l^1) \dots \hat{C}^S(i^S, l^S)]^T \quad (15)$$

优化目标式(14)的求解为凸优化问题, 解为优化目标式(14)的驻点。令 $F(\{z^s\}_{s=0}^S)$ 表示式(14)中的优化目标函数, 如式(16)求偏导数可得式(17)。

$$\frac{\partial F}{\partial z^s} = 0, (0 \leq s < S) \quad (16)$$

$$\begin{cases} (1 + \lambda)z^0 - \lambda z^1 = \tilde{C}^0(i^0, l^0), & s = 0 \\ -\lambda z^{s-1} + (1 + 2\lambda)z^s - \lambda z^{s+1} = \tilde{C}^s(i^s, l^s), & (1 \leq s \leq S-1) \\ -\lambda z^{s-1} + (1 + \lambda)z^s = \tilde{C}^S(i^S, l^S), & s = S. \end{cases} \quad (17)$$

图像 $S+1$ 个高斯金字塔分解尺度对应 $S+1$ 个线性方程, 式(17)方程组可以使用矩阵形式如式(18)表示:

$$\mathbf{A}\hat{\mathbf{v}} = \tilde{\mathbf{v}} \quad (18)$$

其中 $\mathbf{A}$ 为 $(S+1) \times (S+1)$ 的三对角线矩阵, $\mathbf{A}$ 为可逆矩阵, 可得式(19):

$$\hat{\mathbf{v}} = \mathbf{A}^{-1}\tilde{\mathbf{v}} \quad (19)$$

由式(19)可求得最终融合了多个尺度的代价聚合结果, 引入了尺度间的正则化后, 来自粗尺度的匹配代价会修正最精细尺度的匹配代价, 只需使用式(20)求出最精细尺度上的匹配代价 $\hat{C}^0(i^0, l^0)$, 在最精细尺度上使用 WTA 算法即可求得视差图。

$$\hat{C}^0(i^0, l^0) = \sum_{s=0}^S A^{-1}(0, s) \tilde{C}^s(i^s, l^s) \quad (20)$$

1.4 视差精细化

由金字塔变换跨尺度代价聚合框架提取到的视差图 D^{INITIAL} 有噪声, 场景中物体的边缘轮廓信息保留不好。在视差图精细化阶段, 使用彩色图像和提取到的对应视差图做输入, 采用 Matsuo 等人在文献[13]中提出的加权联合双边滤波(weighted joint bilateral filter: WJBF)视差图精细化方法对立体匹配得到的视差图 D^{INITIAL} 进行精细

化, 去除视差图中的噪声, 以及物体边缘轮廓的模糊, 得到质量更好的视差图 D^{WJBF} 。

$$D_p = \frac{\sum_{s \in N} w(p, s) c(I_p, I_s) R_s D_s}{\sum_{s \in N} w(p, s) c(I_p, I_s) R_s},$$

$$w(p, s) = e^{-\frac{1}{2} \left(\frac{\|p-s\|^2}{\sigma_s} \right)}$$

$$c(I_p, I_s) = e^{-\frac{1}{2} \left(\frac{\|I_p - I_s\|^2}{\sigma_c} \right)}$$
(21)

如式(21)所示加权联合双边滤波是在联合双边滤波的基础上加上权值, 其中: p 表示当前像素点; s 为以 p 点为中心的支持窗口内的其他像素点; I 为输入的彩色图像; D 为彩色图像对应的视差图; N 为支持窗口内的像素点个数; w 为基于空间距离的权值, 像素点间空间距离越近, 权值越大, 其中 σ_s 为空间高斯分布参数; c 为基于颜色距离的权值, 像素点间的像素值越相似, 权值越大, 其中 σ_c 为颜色高斯分布参数; R_s 为基于彩色图像和视差图的权值图, 由式(22)决定。

$$R_s = \sum_{q \in N} M_s \cdot w(s, q) c(s, q) d(s, q),$$

$$d(s, q) = e^{-\frac{1}{2} \left(\frac{\|D_s - D_q\|^2}{\sigma_r} \right)}$$
(22)

其中: M_s 为斑点掩码, 斑点为视差图中的噪声区域, 在初始的视差图上使用斑点检测算法得到, 斑点检测算法使用预先设定好的斑点区域大小 `speckleWindowSize` 以及斑点区域内像素点间视差值差异大小 `speckleRange` 两个参数来检测斑点区域, 当视差图上区域内的像素点视差值差异大于 `speckleRange` 且这样的区域尺寸小于 `speckleWindowSize` 时将该区域内的像素点掩码标为 0, 否则像素点的掩码置为 1; w 为基于空间距离的权值; c 为基于颜色距离的权值; d 为基于视差距离的权值, 视差差异越小, d 值越大; N 为支持窗口内的像素点个数。在权值图 R_s 上, 场景中物体的边缘轮廓部分、以及视差图内的噪声区域权值小, 通过使用权值图 R_s 和加权双边滤波来忽略场景中物体边缘部分、以及视差图中噪声部分的视

差值, 最终能够修正视差图物体边缘部分的模糊并减少视差图中的噪声。

使用加权联合双边滤波能够修正视差图中物体边缘轮廓部分的视差值并对视差平坦区域进行平滑, 但在前景和背景颜色差异小、前景和背景视差值差异大的部分会引入轻微的模糊, 对这些区域的视差值 D^{WJBF} 需要使用式(23)进行修正补偿, 其中 $D_p^{\text{COMPENSATORY}}$ 为补偿后的视差值, W 为以 p 点为中心的支持窗口。

$$D_p^{\text{COMPENSATORY}} = D_p^{\text{INITIAL}}$$

$$v = \arg \min_{s \in W} \|D_p^{\text{WJBF}} - D_s^{\text{INITIAL}}\|^2$$
(23)

最后, 总结本文算法整体伪代码如下:

金字塔变换跨尺度代价聚合视差图提取流程

输入: 立体图像对 I, I'

1. 建立高斯金字塔,
 2. 使用式(2)计算每个尺度下的匹配代价卷
 3. 使用式(12)计算每个尺度下经过拉普拉斯上采样匹配代价卷的代价聚合
 4. 使用式(20)计算最精细尺度下匹配代价卷
 5. 在最精细尺度下的匹配代价卷上使用 WTA 算法计算视差图
 6. 使用视差图精细化算法优化得到的视差图
- 输出: 图像 I 的视差图

2 实验结果与分析

论文使用 Middlebury^[14]提供的立体匹配数据集来验证算法的有效性。实验环境为: Visual Studio 2013 下 C++编程环境, Windows 7 X86 操作系统, Intel(R) Core(TM) i7-3770 3.4 GHz, 8 GB 内存。

图 2 和 3 给出了两组 Zhang 等人论文得到的视差图、AdaptWeigh^[15]算法得到的视差图、本论文得到的视差图以及 Middlebury Stereo Vision 提供的精准视差图, 表 1 给出了立体图像对求得的视差图在非遮挡区域(nonocc)、视差不连续区域(disc)、所有区域(all)在错误阈值为 1.0 时的匹配错误率, 可以看出在这 3 种区域本论文的方法相比 Zhang 等人的方法都能减少立体匹配的错误率。

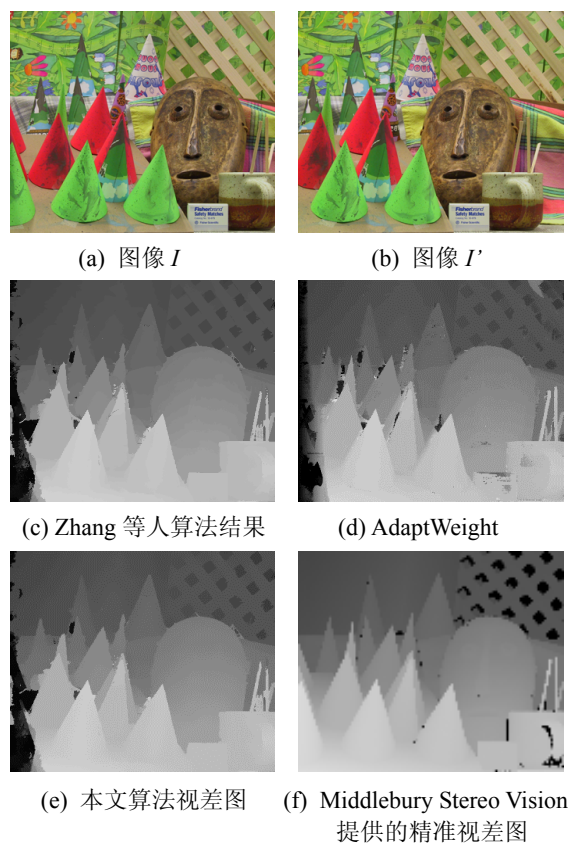


图 2 Cones 立体图像对计算得到的视差图

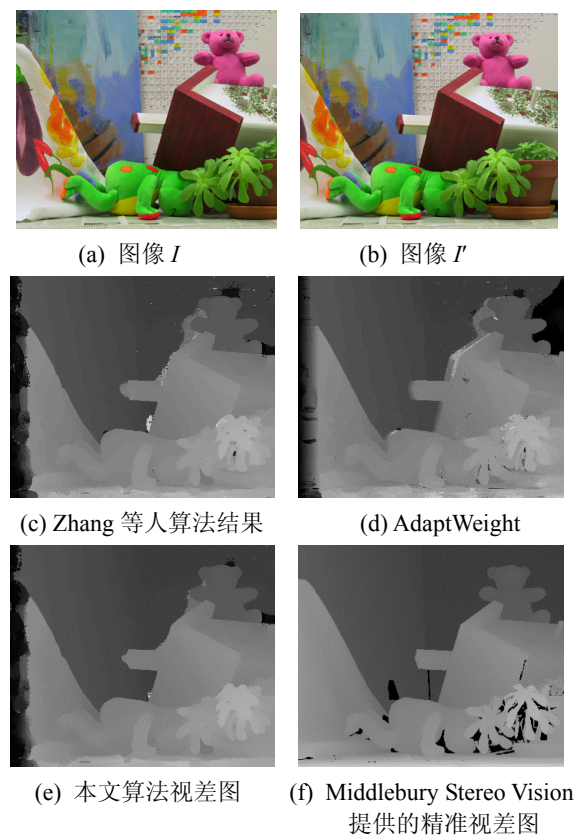


图 3 Teddy 立体图像对计算得到的视差图

表 1 立体匹配错误率

数据	方法	nocc	all	disc
cones	Zhang 等人方法	3.51	10.71	9.18
	本文方法	3.01	9.90	8.43
teddy	Zhang 等人方法	6.64	13.97	17.48
	本文方法	5.94	13.23	16.09
tsukuba	Zhang 等人方法	4.86	5.61	22.87
	本文方法	4.33	4.99	20.23
venus	Zhang 等人方法	1.45	2.50	11.62
	本文方法	0.99	1.81	8.75

3 结论

论文改进了一种基于高斯金字塔变换跨尺度代价聚合的立体匹配方法,在原有高斯金字塔跨尺度代价聚合框架基础上在代价聚合阶段加入拉普拉斯金字塔变换,在精细化阶段加入基于加权联合双边滤波的视差图精细化,新的跨尺度代价聚合与视差精细化框架能获得的比原论文质量更好的视差图。本文算法在遮挡区域无法求取视差值,下一步计划使用基于边缘轮廓信息的稠密插值方法,使用同一个物体非遮挡区域的视差值,去估算遮挡区域的视差值,以此来给遮挡区域进行近似插值,计算出遮挡区域的视差,进一步提高视差图的精度。

参考文献:

- [1] 塞利斯基 (Szeliski, R.). 计算机视觉 [M]. 北京: 清华大学出版社, 2012: 410.
- [2] 曹之乐, 严中红, 王洪. 双目立体视觉匹配技术综述 [J]. 重庆理工大学学报(自然科学版), 2015, 29(2): 70-75.
- [3] Scharstein D, Szeliski R. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms [C]// Stereo and Multi-Baseline Vision. USA: IEEE, 2001: 131-140.
- [4] Yoon K J, Kweon I S. Adaptive Support-Weight Approach for Correspondence Search [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence (S0162-8828), 2006, 28(4): 650-656.
- [5] Hosni A, Rhemann C, Bleyer M, et al. Fast Cost-Volume Filtering for Visual Correspondence and Beyond [J]. IEEE Transactions on Software Engineering (S0098-5589), 2013, 35(2): 504-511.

(上接第 2226 页)