

# 自适应权值的跨尺度立体匹配算法

李培玄<sup>1,2,3,4</sup>, 刘鹏飞<sup>1,2,3,4</sup>, 曹飞道<sup>1,2,3,4</sup>, 赵怀慈<sup>1,3,4\*</sup>

<sup>1</sup>中国科学院沈阳自动化研究所, 辽宁 沈阳 110016;

<sup>2</sup>中国科学院大学, 北京 100049;

<sup>3</sup>中国科学院光电信息处理重点实验室, 辽宁 沈阳 110016;

<sup>4</sup>辽宁省图像理解与视觉计算重点实验室, 辽宁 沈阳 110016

**摘要** 现有的多尺度立体匹配算法对各尺度的代价函数采用相同权值, 而忽略了各尺度层对整个匹配代价的不同影响, 增加了误匹配点。针对此问题, 提出了自适应权值的跨尺度立体匹配算法框架。采用统一的代价聚合函数框架在不同尺度上进行代价匹配, 并提出利用各像素窗口的信息熵作为不同尺度下匹配代价对整个匹配代价的影响因子; 同时为了保证不同尺度下同一像素的代价一致性, 在代价函数里加入正则化因子。本文算法框架可以应用在利用多尺度进行代价匹配的算法上, 并使原有算法的准确率和稳健性得到提高。基于本文算法框架, 分别采用不同代价聚合函数在 Middlebury 数据集上进行测试。为保证测试的公平性, 各算法均未进行后续的视差求精步骤, 实验表明, 本文算法有效地提高了多尺度立体匹配的准确率和稳健性。

**关键词** 机器视觉; 立体匹配; 跨尺度代价聚合; 图像信息熵; 自适应权值

中图分类号 TN911.73

文献标识码 A

doi: 10.3788/AOS201838.1215006

## Weight-Adaptive Cross-Scale Algorithm for Stereo Matching

Li Peixuan<sup>1,2,3,4</sup>, Liu Pengfei<sup>1,2,3,4</sup>, Cao Feidao<sup>1,2,3,4</sup>, Zhao Huaici<sup>1,3,4\*</sup>

<sup>1</sup>Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, Liaoning 110016, China;

<sup>2</sup>University of Chinese Academy of Sciences, Beijing 100049, China;

<sup>3</sup>Laboratory of Opto-Electronic Information Processing, Chinese Academy of Sciences, Shenyang, Liaoning 110016, China;

<sup>4</sup>Key Laboratory of Image Understanding and Computer Vision, Shenyang, Liaoning 110016, China

**Abstract** In the existing weight-adaptive cross-scale algorithms, the same weight for the cost function of each scale is adopted, the different influence of each scale layer on the whole matching cost is missing, and thus the number of mismatching points increases. As for this problem, a weight-adaptive cross-scale algorithm framework for stereo matching is proposed. The cost matching is performed on different scales in the framework of unified cost aggregation function and the information entropy of each pixel window is used as the influence factor of the matching cost at each scale on the whole matching cost. At the same time, a regularization factor is added to the cost function to ensure the cost consistency at different scales for the same pixel. The proposed algorithm framework can be applied to the multi-scale algorithm of cost matching and improve the accuracy and robustness of the existing algorithms. Based on the proposed algorithm framework, the different cost aggregate functions are tested on the Middlebury dataset. To ensure the fairness of tests, as for each algorithm, there is no a subsequent parallax refinement step. The experimental results show that the proposed algorithm effectively improves the accuracy and robustness of multi-scale stereo matching.

**Key words** machine vision; stereo matching; cross-scale cost aggregation; entropy of image information; adaptive weight

**OCIS codes** 150.1135; 330.1400; 100.6890

收稿日期: 2018-06-12; 修回日期: 2018-07-23; 录用日期: 2018-07-27

基金项目: 装备预研领域基金(61400010102)

\* E-mail: hczhao@sia.cn

## 1 引 言

立体匹配是计算机视觉中的关键性问题<sup>[1]</sup>,旨在寻找多幅图像中相对应的点,广泛应用于机器人导航、无人驾驶、虚拟现实、三维重建等领域<sup>[2]</sup>。立体匹配技术的研究热点主要集中于基于局部、基于全局和基于半全局的立体匹配算法<sup>[3]</sup>。Scharstein等<sup>[4]</sup>将立体匹配算法分为4个步骤:代价计算、代价聚合、视差计算和视差求精。对于代价计算阶段,在可能的视差范围内,根据对应像素的匹配代价生成三维代价体。对于代价聚合阶段,在每个像素的窗口区域内,根据相应的核函数进行代价聚合。然后每个像素的视差可以由局部或者全局优化方法求得。最后对视差图进行求精计算。在以上步骤里,代价聚合为求取精确视差图的关键步骤<sup>[5-8]</sup>,因此本文主要针对代价聚合进行改进。

大多数代价聚合可以看成是对代价体进行滤波操作<sup>[6]</sup>。盒滤波和高斯滤波为最基本的线性滤波器,运行速度较快,但其会模糊视差图像使精度下降<sup>[4]</sup>。Yoon等<sup>[8]</sup>将具有边缘感知的双边滤波器应用在代价聚合步骤上,通常需要较大的滤波窗口才能生成比较准确的视差图,计算复杂度较大。Hosni等<sup>[6]</sup>把引导滤波器应用在代价聚合上,其计算复杂度与窗口大小无关<sup>[9]</sup>。Yang等<sup>[7]</sup>提出一种半局部代价聚合方法,其将整幅图像作为核函数窗口,利用最小生成树加速代价匹配。Mei等<sup>[5]</sup>先对图像进行分割,再采用半局部方法对每个分割块求取子树,然后根据贪心算法,将各个分割对应的子树进行合并。这些算法极大提高了立体匹配的精度,但其未利用像人眼视觉系统采取由粗到精的策略融合多尺度图像信息来处理匹配问题<sup>[10-11]</sup>,无法充分利用图像内容信息。因此,Zhang等<sup>[12]</sup>提出跨尺度代价聚合方法,并设计了统一的代价聚合模型。不仅融合了现有的多种立体匹配算法,还考虑了跨尺度交互、引入正则化项,目前被多数多尺度匹配算法所采用。

综上所述,聚合算法通常在多尺度下进行代价聚合,其根据人眼视觉模型融合不同尺度的匹配代价,从而减小误匹配率。但其均简单地将各个尺度图像匹配代价进行相加,未考虑由于高斯滤波和下采样造成的信息丢失,使得单一尺度下匹配正确的点在多尺度融合中出现错误匹配。因此,本文引入信息熵计算每个像素窗口在各尺度层的信息量,作为每个像素在各尺度对整个匹配

代价的影响因子,并且形成自适应权值的跨尺度代价聚合统一算法框架。在Middlebury数据集<sup>[13]</sup>上分别对盒子滤波、双边滤波、半局部代价聚合、基于图割的半局部代价聚合和引导滤波等代价聚合方法进行评价。结果表明,在少量增加计算量的情况下,所提算法可以提高求取视差的精度和稳健性。

## 2 算法描述

所提算法由代价计算、代价聚合、视差计算等步骤组成。

### 2.1 代价计算

代价计算用来计算不同视图下对应像素点的匹配关系。一般用  $f: \mathbf{R}^{W \times H \times 3} \times \mathbf{R}^{W \times H \times 3} \mapsto \mathbf{R}^{W \times H \times L}$  表示,其中  $W$ 、 $H$  为输入图像的长和宽,3 为图像颜色通道, $L$  为两图像对应像素点的最大视差  $d_{\max}$ 。因此两幅图像对  $I, I' \in \mathbf{R}^{W \times H \times 3}$  的匹配代价可表示为

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

由(1)式可以计算出整幅图像的代价体  $C \in \mathbf{R}^{W \times H \times L}$ ,代表每个像素在最大视差下的匹配代价。每一个像素  $i = (x_i, y_i)$  在视差  $l$  下的匹配代价表示为  $C(i, l)$ 。将颜色与梯度信息相结合计算匹配代价<sup>[5-7]</sup>:

$$C(i, l) = (1 - \alpha) \cdot \min[\|I(i) - I'(i_l)\|, \tau_1] + \alpha \cdot \min[\|\nabla_x I(i) - \nabla_x I'(i_l)\|, \tau_2], \quad (2)$$

式中: $I(i)$ 为像素  $i$  颜色的三维列向量; $\nabla_x$ 为相应图像转化为灰度图像后在  $x$  方向的梯度; $i_l = (x_i - l, y_i)$ 为像素在视差为  $l$  时对应的像素; $\alpha$ 为颜色信息和梯度信息的平衡权值; $\tau_1$ 、 $\tau_2$ 为截断值。

### 2.2 代价聚合

代价聚合通常采用传统代价聚合与多尺度相结合的方法,在少纹理或无纹理区域可以增加匹配准确度。但在跨尺度代价聚合时,上述方法将各尺度匹配代价对整个匹配代价的影响同等对待,即影响因子为1。而由于高斯模糊和下采样后某些有纹理区域在粗尺度上可能转化为无纹理区域,此时仍利用相同权值的该粗尺度匹配代价将造成错误匹配,增大整个匹配代价的误差。如图1所示,图1(a)为原图像,图1(b)为第二尺度层图像。由于高斯模糊和下采样后蓝色方框中原本有纹理的区域变为了光滑无纹理区域。此时采用文献<sup>[12]</sup>的多尺度代价聚合方法的结果在黑色方框内表示,红色为错误匹配点,可见在此处利用相同权值加入粗尺度层会增加误匹配点。因此在利用多尺度进行代价

聚合时,应考虑像素窗口对整个匹配代价的影响。当某像素窗口内纹理较少时应减少该窗口对整个匹配代价的权值大小,相应地,当某像素窗口内纹理较丰富时增大权值大小。本文基于此利用图像信息熵来表述像素窗口纹理丰富程度,然后利用归一化的熵值大小来表述各尺度匹配代价对整个匹配代价的影响因子。最后为了提高同一像素多尺度下的一致性,本文加入了平滑性约束更强的二范数正则化项,从而减少跨尺度代价聚合时对原本有纹理区域的影响。

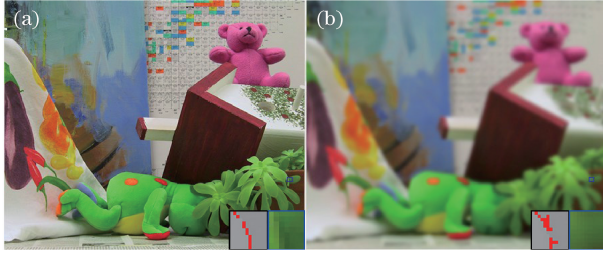


图 1 利用不同尺度进行代价聚合时某像素窗口的图像(蓝色方框内)和误匹配点(黑色方框内)。(a)原图像;(b)第二尺度层图像

Fig. 1 Image (blue box) and mismatching points (black box) of certain pixel window when cost aggregation performed at different scales. (a) Original image; (b) second-scale image

### 2.2.1 图像信息熵

图像信息熵是一种特征表述形式,可以用来描述图像的平均信息量,数学上表示为信息量的期望值,其在图像中的计算公式为

$$E = - \sum_{x \in U} P(x) \ln P(x), \quad (3)$$

式中  $P(x)$  为图像中灰度为  $x$  的像素的概率,即在整幅图像灰度所占的比例,  $U \in [0, 255]$  为图像的灰度范围。如图 2 所示,当图像全为黑色或全白时,  $E=0$ ,当图像为黑白相间时,  $E>0$ ,更为密集的黑白相间图像信息熵更大。因此,当窗口纹理比较丰富即信息量期望值较大时图像信息熵也相应变大,窗口纹理较少时图像信息熵也相应变小。

### 2.2.2 自适应权值的多尺度代价聚合

自适应权值的多尺度代价聚合是将信息熵引入多尺度代价聚合统一框架内,并使其表示某像素窗口在各尺度下的影响因子,最后加入正则项去统一该像素的匹配一致性。

首先将图像连续高斯平滑并以采样因子为  $\eta$  下采样,得到不同尺度图像。并将其引入多尺度空间的统一聚合框架,公式如下:

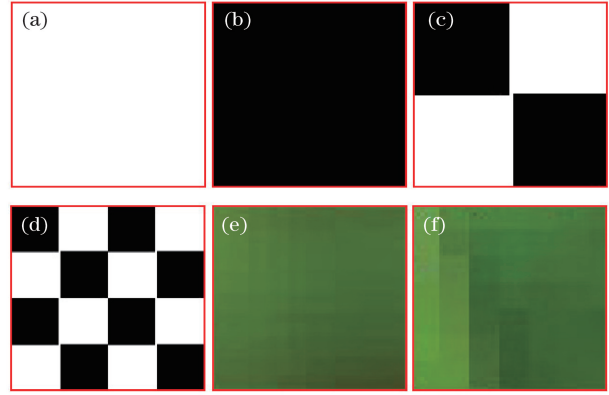


图 2 不同图像及其对应的信息熵。(a)纯白色图像  $E=0$ ; (b)纯黑色图像  $E=0$ ; (c)4 方格图像  $E=1.0413$ ; (d)16 方格图像  $E=1.3476$ ; (e)图 1(b)中蓝色方框内图像  $E=5.0542$ ; (f)图 1(a)中蓝色方框图像  $E=5.6215$

Fig. 2 Different images and corresponding information entropy. (a)  $E=0$  for pure white image information; (b)  $E=0$  for pure black image; (c)  $E=1.0413$  for 4-grid image; (d)  $E=1.3476$  for 16-grid image; (e)  $E=5.0542$  for image in blue box of Fig. 1(b); (f)  $E=5.6215$  for image in blue box of Fig. 1(a)

$$\tilde{v} = \arg \min_{\{z^s\}_{s=0}^S} \sum_{s=0}^S \frac{1}{Z^s} \sum_{j^s \in N_{i^s}} K(i^s, j^s) \|z^s - C^s(j^s, l^s)\|^2, \quad (4)$$

式中,  $s \in \{0, 1, \dots, S\}$  为不同尺度;  $K(i, j)$  为像素  $i, j$  匹配程度的相似核<sup>[7]</sup>,可以根据要求利用不同的核函数或者滤波器进行代价聚合,例如盒子滤波、双边滤波、引导滤波和基于树结构的核函数等;  $Z_{i^s}^s = \sum_{j^s \in N_{i^s}} K(i^s, j^s)$  为多尺度归一化因子,  $\{i^s\}_{s=0}^S$  和  $\{l^s\}_{s=0}^S$  为不同尺度的像素和其对应的视差值,并且  $i^{s+1} = i^s / \eta, l^{s+1} = l^s / \eta; N_{i^s}$  为不同尺度图像下对应像素的聚合窗口大小,该窗口可以根据不同尺度进行更改,也可以保持不变使其在粗尺度下利用更多的邻域信息;  $\tilde{v} = [\tilde{C}(i^0, l^0), \tilde{C}(i^1, l^1), \dots, \tilde{C}(i^s, l^s)]$  为  $S+1$  个尺度下各尺度的匹配代价。(4)式的解相当于在各尺度下分别求对应的匹配代价,表示为

$$\forall s, \tilde{C}^s(i^s, l^s) = \frac{1}{Z_{i^s}^s} \sum_{j^s \in N_{i^s}} K(i^s, j^s) C^s(j^s, l^s). \quad (5)$$

将归一化的图像信息熵作为各尺度匹配代价对整个匹配代价的影响因子,同时为了保证各像素在多尺度下的一致性和减少噪声干扰,加入二范数正

则化项。其公式为

$$\tilde{v} = \arg \min_{\{z^s\}_{s=0}^S} \left[ \sum_{s=0}^S \frac{E_{N_{i^s}}}{Z_{i^s}^S} \sum_{j^s \in N_{i^s}} K(i^s, j^s) \|z^s - C^s(j^s, l^s)\|^2 + \lambda \sum_{s=1}^S \|z^s - z^{s-1}\|^2 \right], \quad (6)$$

式中  $E_{N_{i^s}}$  为  $S$  尺度下  $i$  像素聚合窗口的图像信息熵。可知(4)式为凸函数,因此通过求方程驻点可以得到局部最优值。令  $\tilde{F}(\{z^s\}_{s=0}^S)$  表示优化项,当  $s \in \{1, 2, \dots, S-1\}$  时,对各尺度匹配代价的偏导数为

$$\begin{aligned} \frac{\partial \tilde{F}}{\partial z^s} &= \frac{2E_{N_{i^s}}}{Z_{i^s}^S} \sum_{j^s \in N_{i^s}} K(i^s, j^s) [z^s - C^s(j^s, l^s)] + \\ &2\lambda(z^s - z^{s-1}) - 2\lambda(z^{s+1} - z^s) = \\ &2[-\lambda z^{s-1} + (E_{N_{i^s}} + 2\lambda)z^s - \lambda z^{s+1} - E_{N_{i^s}}\tilde{C}^s(i^s, l^s)]. \end{aligned} \quad (7)$$

令  $\frac{\partial \tilde{F}}{\partial z^s} = 0$  得:

$$-\lambda z^{s-1} + (E_{N_{i^s}} + 2\lambda)z^s - \lambda z^{s+1} = E_{N_{i^s}}\tilde{C}^s(i^s, l^s). \quad (8)$$

因此对于  $s \in \{0, 1, \dots, S\}$ , 可得:

$$\begin{cases} (E_{N_{i^s}} - \lambda)z^s + \lambda z^{s+1} = E_{N_{i^s}}\tilde{C}^s(i^s, l^s), s=0 \\ -\lambda z^{s-1} + (E_{N_{i^s}} + 2\lambda)z^s - \lambda z^{s+1} = E_{N_{i^s}}\tilde{C}^s(i^s, l^s), \\ \quad s = [1, 2, \dots, S-1] \\ (E_{N_{i^s}} + \lambda)z^s - \lambda z^{s-1} = E_{N_{i^s}}\tilde{C}^s(i^s, l^s), s=S \end{cases}. \quad (9)$$

对于  $S+1$  个尺度,可以将  $S+1$  个线性方程组表述为矩阵形式:

$$\mathbf{A}\tilde{v} = \begin{bmatrix} E_{N_{i^0}}\tilde{C}^0(i^0, l^0) \\ E_{N_{i^1}}\tilde{C}^1(i^1, l^1) \\ \vdots \\ E_{N_{i^{S+1}}}\tilde{C}^{S+1}(i^{S+1}, l^{S+1}) \end{bmatrix}, \quad (10)$$

式中  $\mathbf{A}$  为  $(S+1) \times (S+1)$  的可逆系数矩阵,因此各尺度的匹配代价为

$$\tilde{v} = \mathbf{A}^{-1} \begin{bmatrix} E_{N_{i^0}}\tilde{C}^0(i^0, l^0) \\ E_{N_{i^1}}\tilde{C}^1(i^1, l^1) \\ \vdots \\ E_{N_{i^{S+1}}}\tilde{C}^{S+1}(i^{S+1}, l^{S+1}) \end{bmatrix}. \quad (11)$$

然后在最细尺度图像下生成最终的匹配代价:

$$\tilde{C}^0(i^0, l^0) = \sum_{s=0}^S \mathbf{A}^{-1}(0, s) E_{N_{i^s}} \tilde{C}^s(i^s, l^s). \quad (12)$$

最后采用 Winner-Takes-All 策略获得最终的视差图像。

### 3 实验结果与分析

本文采用统一的 Middlebury 数据集进行测试,计算机配置为 Windows 7 x64 系统、Intel Core i7-6700 3.4 GHz 主频 CPU、16 GB 内存。

算法部分参数参照文献[12]设置,如表 1 所示。

表 1 本文跨尺度代价匹配算法参数  
Table 1 Parameters for proposed cross-scale cost matching algorithm

Parameter	$\tau_1$	$\tau_2$	$\alpha$	$\eta$	$S$	$\lambda$
Value	0.02745	0.00784	0.11	2	4	0.27

为了验证本文算法的效果,采用 Middlebury 数据集中的 18 幅图像进行测试,不仅包含 Teddy、cones、dolls 等纹理丰富图片,还包含 Bowling2、Lampshade2、Middl 等纹理较少甚至无纹理图像,更能反映算法的好坏。按照 Middlebury 的统一评价标准,为更好体现本文提出算法的精确度,设置误差阈值为 1。先分别采用盒子滤波(BOX)、双边滤波(BF)、基于图割的半局部聚合方法(ST)、半局部聚合方法(NL)、引导滤波(GF)对代价体进行聚合,对文献[12]方法用“S+”作为前缀表示,本文方法用“AS+”作为前缀表示,其分别对 Middlebury 中 Teddy 图像对求取视差,为了保证公平性所有方法均未采用视差求精步骤。效果如图 3 所示,图中红色表示所求视差值与标准视差比较误差大于 1 的位置,蓝色为遮挡区域,括号内为其非遮挡区域的坏点率。

从图 3 可以看出,文献[12]方法均能提高传统方法的匹配性能,而本文方法对文献[12]方法进一步改进从而减少了坏点率。再利用 Middlebury 的 18 幅图像在误差阈值为 2 时,分别用盒子滤波进行比较,结果如表 2 所示。

由表 2 可知,本文算法在图像非遮挡区域平均误匹配率为 16.34%,使用盒子滤波后,本文算法的匹配误差比原有的跨尺度算法也降低了 1.63%。

目前常用聚合算法作为具有边缘感知的引导滤波,其由于时间复杂度与窗口大小无关,因此运行效率较快,匹配误差也较小。为此测试了文献[12]方



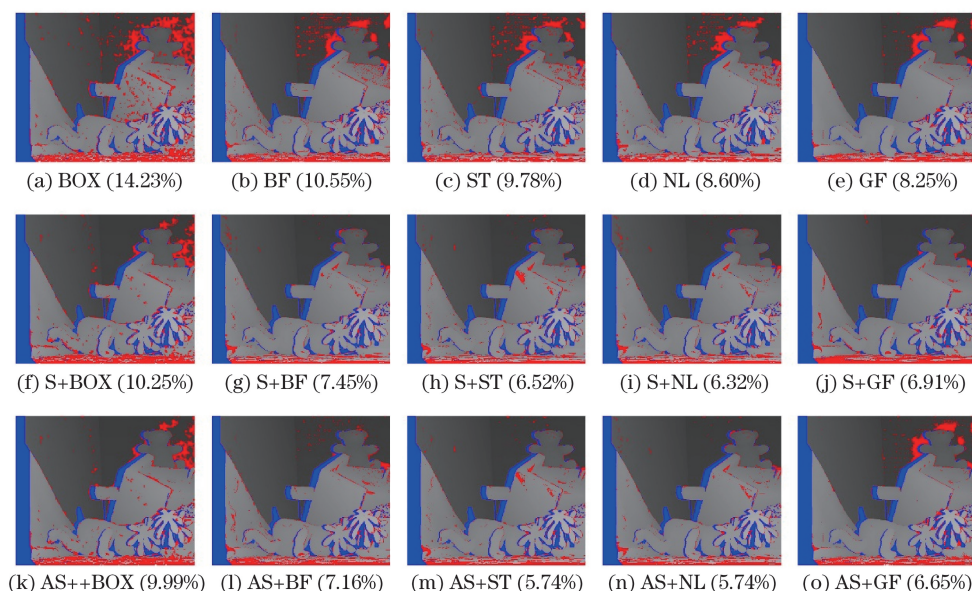


图 3 不同方法对 Teddy 图像求取的视差图

Fig. 3 Parallax maps of Teddy images by different methods

表 2 坏点率为 2 时本文方法与文献[12]方法的匹配误差对比

Table 2 Matching error comparison between proposed method and that in Ref. [12] when dead pixel rate is 2 %

Stereo Pairs	S+BOX		AS+BOX	
	Non-occluded	All	Non-occluded	All
Bowling2	10.07	22.46	9.43	21.64
Baby1	10.92	15.36	8.35	12.68
Cloth3	7.42	11.66	7.01	11.00
Flowerpots	15.59	30.78	14.43	29.75
Lampshade2	30.41	36.58	24.72	31.33
Midd1	46.53	49.43	42.00	45.16
Monopoly	37.08	42.48	25.44	31.78
Plastic	56.12	57.38	55.47	56.77
Rocks1	12.04	17.42	11.94	17.31
Wood1	13.11	19.25	12.75	18.92
Books	15.51	22.22	14.91	21.68
Moebius	16.82	22.64	16.52	22.36
Dolls	7.51	14.72	7.51	14.75
Baby2	6.88	12.12	6.93	12.18
Wood2	16.77	18.22	16.80	18.25
Rocks2	8.79	15.14	8.82	15.19
Teddy	7.18	16.13	7.13	16.01
Cones	3.99	13.56	3.86	13.52
Average	17.93	24.31	16.34	22.80

法与本文算法在利用引导滤波进行代价聚合时的运行时间对比。结果如表 3 所示。

表 3 利用引导滤波进行代价聚合时运行时间对比

Table 3 Running time comparison for cost aggregation by guiding filters

Stereo pairs	S+GF	AS+GF
Bowling2	6.75220	7.32107
Baby1	5.43631	5.54525
Cloth3	6.28890	6.33252
Flowerpots	6.56278	6.68801
Lampshade2	7.53322	7.98634
Midd1	6.96293	7.36999
Monopoly	6.57855	6.82937
Plastic	6.94070	7.30089
Rocks1	6.17969	6.27993
Wood1	6.43060	6.46233
Books	6.30772	6.76049
Moebius	6.03519	6.98040
Dolls	5.91785	6.18341
Baby2	5.18777	5.40952
Wood2	6.26264	6.80027
Rocks2	5.78323	5.95239
Teddy	5.55470	6.00646
Cones	5.56898	5.85602
Average	6.237998	6.559148

由表 3 可知,由于加入了对每个像素窗口求取图像信息熵的步骤,本文算法较文献[12]方法在匹配时间上略有增加,但仅为毫秒级影响,在实际应用中可忽略不计。

## 4 结 论

针对传统多尺度算法利用相同权值对各尺度进行代价聚合造成错误匹配,提出了自适应权值的多尺度立体匹配算法。引入图像信息熵,以各像素对应窗口在各尺度的信息熵大小作为各尺度对整个匹配代价的贡献程度,充分利用图像信息,减少了原有算法由于高斯模糊和下采样造成的错误匹配点。加入正则化项进行各尺度一致性约束,进一步降低了误匹配率。将所提算法与原有算法进行实验对比,在损失可忽略不计的运行时间下,本文算法有效降低了错误匹配率。在后续的工作中,将利用其他窗口纹理评价算法生成各尺度影响因子,在基本不影响运行时间的情况下,进一步降低错误匹配率。

## 参 考 文 献

- [1] Liu C, Yuen J, Torralba A. SIFT flow: dense correspondence across scenes and its applications[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(5): 978-994.
- [2] 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.
- [3] Lin S, Yin X Y, Tang Y D. Research status and prospect of binocular stereo matching technology[J]. Science Technology and Engineering, 2017, 17(30): 135-147.  
林森, 殷新勇, 唐延东. 双目视觉立体匹配技术研究现状和展望[J]. 科学技术与工程, 2017, 17(30): 135-147.
- [4] Scharstein D, Szeliski R, Zabih R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms[C]//Proceedings of IEEE Workshop on Stereo and Multi-Baseline Vision, 2001: 131-140.
- [5] Mei X, Sun X, Dong W M, *et al.* Segment-tree based cost aggregation for stereo matching[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013: 313-320.
- [6] Rhemann C, Hosni A, 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.
- [7] Yang Q X. A non-local cost aggregation method for stereo matching[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2012: 1402-1409.
- [8] Yoon K J, Kweon I S. Adaptive support-weight approach for correspondence search[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(4): 650-656.
- [9] Liu J, Zhang J X, Dai Y, *et al.* Dense stereo matching based on cross-scale guided image filtering[J]. Acta Optica Sinica, 2018, 38(1): 0115004.  
刘杰, 张建勋, 代煜, 等. 基于跨尺度引导图像滤波的稠密立体匹配[J]. 光学学报, 2018, 38(1): 0115004.
- [10] Mallot H A, Gillner S, Arndt P A. Is correspondence search in human stereo vision a coarse-to-fine process?[J]. Biological Cybernetics, 1996, 74(2): 95-106.
- [11] Marr D, Poggio T. A computational theory of human stereo vision[J]. Proceedings of the Royal Society of London. Series B, Biological Sciences, 1979, 204(1156): 301-328.
- [12] Zhang K, Fang Y Q, Min D B, *et al.* Cross-scale cost aggregation for stereo matching[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2014: 1590-1597.
- [13] Scharsteinand D, Szeliski R, Hirschmüller H. The middlebury stereo vision page[EB/OL]. [2018-06-12]. <http://vision.middlebury.edu/stereo/>.