基于互信息的立体匹配方法

杜思傲, 尹业安*, 吴文俊

(武汉纺织大学 数学与计算机学院,湖北 武汉 430200)

摘 要: 介绍一种基于灰度互信息的立体匹配方法,用于从双目图中获得视差图。该方法将灰度互信息作为衡量 输入图像对相似度的指标,先构建单像素点的代价,再在代价聚合环节加入邻域点的约束,使用多方向的扫描线 优化来优化代价聚合函数。反复地进行优化过程,让输出的视差图渐渐地接近于真实情况,使用随机图作为迭代 初始状态,迭代固定次数后输出。灰度互信息作为代价标准,使得该方法在代价的计算环节比一般的全局匹配方 法更快,并对光照具有一定鲁棒性,领域点的约束让视差图更加平滑稠密。

文献标识码:A

关键词: 立体匹配; 互信息; 代价; 优化

中图分类号: TP391.4

文章编号: 2095-414X(2019)02-0057-08

1 引言

获取场景深度是计算机视觉的重要课题之一,获得的深度图可以用于三维重建、SLAM、汽车导航、 目标跟踪、非接触测量等。结构光、光度立体、SFM、双目立体视觉都是目前主流的方法,双目立体视觉 相对于其他方法,具有设备简单、实现容易等特点。将两个同型号的摄像头并排固定,保证两个摄像头的 成像平面在同一平面,控制其同时拍照成像,就构建了理想的双目立体视觉系统^{III}。理论上,已知世界坐 标系中某一点在双目图像中的位置,以及相机的内参和外参,计算两个点在像素尺寸上的差值,通过三角 测距原理,就能获取该世界坐标系中点相对于相机的深度^{III}。所以双目立体视觉的核心问题就是如何获取 任一物理点在双目图像中的位置,及立体匹配。立体匹配的方法根据采用的最优化理论方法,主要分为两 大类:局部匹配法和全局匹配法。

局部匹配算法主要对像素点周围小范围区域进行约束,将局部区域匹配代价最小或者匹配度最高的视差值作为输出值,特点是速度快、实现简单、精度高,但是局部匹配算法比较依赖图像中的纹理和特征,不适合低纹理区域的匹配,而且对噪声非常敏感,不具有鲁棒性。

全局匹配算法采用全局优化理论方法估计视差,Zhang Kang 设计了一种一般化的代价聚合模型^[2],先 建立全局的代价函数,再通过最小化全局代价函数得到最优视差值作为输出值,代表性的全局算法有 Graph cuts^[3,4],belief propagation^[5,6],DoubleBP^[7]等。用于建立全局立体匹配的能量函数的方法有很多,包括水平滑 动的模板匹配窗、最小生成树^[8,9]、色彩权重^[7]等,有些全局匹配方法在能量函数的建立上并不需要图像所 有点都参与计算,通常这种做法也被称为也被称为并不需要图像括水平滑动的模板匹配窗、最小生成。

本文将使用灰度互信息构建能量函数,灰度互信息是一种评估匹配程度的标准,具有计算量小而且对 光照具有鲁棒性的特点,优化方法将采用多方向的扫描线优化,多方向的扫描线可以减少扫描线优化所造 成的条纹瑕疵。

2 熵和灰度互信息

2.1 随机变量的熵与条件熵

在信息论中使用熵来表示随机变量的不确定性¹⁰,即变量的信息量,不确定性越强那么熵的值就会越大,连续变量的熵公式如下。

$$H(x) = -\int p(x)\ln p(x) \, dx \tag{1}$$

图像的熵则表示为图像的信息量,在使用255分度的灰度值作为变量值域时,熵表示为离散形式,离 散变量的熵公式如下。

$$H(x) = -\sum_{n=1}^{\infty} \ln p(x) \tag{2}$$

在计算机系统中可以使用大小为 255 的队列来表示不同灰度值的分布情况,即n为 255。

假设有一个联合概率分布p(x,y),从这个概率分布中抽取 一对x和y,如果x的值已知,那么需要确定 对应的y值所需的附加的信息就是-ln p(x,y),因此用于确定y值的平均附加信息可以写成H(y|x)。

$$H(y|x) = - \iint p(y,x) \ln p(y|x) \, dy dx \tag{3}$$

H[y|x]被称为给定 x 的情况下, y 的条件熵, 条件熵满足以下关系。

$$H(x, y) = H(y|x) + H(x)$$

(4)

其中H(x,y)是p(x,y)的微分熵,也是x和y的联合熵,H(x)是边缘分布p(x)的微分熵。在离散的图像系统 中, x和y的变换范围是 0-255 的离散灰度值,联合熵在计算机系统中可以使用大小为 256*256 的图来表示。



(a) 左图

图 1 双目图与联合熵图

2.2 相对熵和互信息

相对熵是也被称为 KL 散度^[10],考虑某个未知的分布p(x),假设已经使用一个近似的分布q(x)对其进 行建模,如果使用q(x)建立一个编码体系,用于把x的值传递给接收者,那么由于使用了q(x)而不是真实 分布p(x),因此在具体化x时,需要一些附加的信息,而需要的平均附加信息量为KL(p||q)。

$$\operatorname{KL}(p||q) = -\int p(x) \ln q(x) \, dx - \left(-\int p(x) \ln p(x) \, dx\right)$$
$$= -\int p(x) \ln \left\{\frac{q(x)}{p(x)}\right\} dx \tag{5}$$

KL 散度是评估两个变量是否相互独立的重要手段,通过求联合概率p(x,y)和边缘概率的乘积 p(x) * p(y)的 KL 散度来判断p和q "相互独立度来判断)变量是否相互独立的重要手段,通过求联合概率使 x与变量y的互信息I(x, y)。

$$I(x,y) = \mathrm{KL}(p(x,y)||p(x) * p(y)) = -\iint p(x,y)\ln(\frac{p(x) * p(y)}{p(x,y)}) \, dx \, dy \tag{6}$$

仅当*I(x,y*)等于 0 的时候x和y相对独立,使用概率的加和规则和乘积规则,可以将互信息简化为熵和 条件熵的形式。

$$I(x, y) = H(x) - H(x|y) = H(y) - H(y|x)$$
(7)

可以把互信息看成由于知道 v 值而造成的 x 的不确定性的减小, 带入条件熵的公式, 则获取互信息的 熵和联合熵的表达式。

$$I(x, y) = H(x) + H(y) - H(x, y)$$
(8)

计算两幅图的灰度互信息是比较两幅图相似程度的重要指标,灰度互信息可以在计算机系统中构建 256*256 的查询表,所以使用灰度互信息构建每个像素的代价函数可以从 256*256 的表中查询对应的代价 值,极大地降低了计算成本。

代价函数与代价聚合 3

连续系统的熵和联合熵为积分形式。

$$H_{I} = -\int_{0}^{1} P_{I}(i) \ln P_{I}(i) \, di \tag{9}$$

$$H_{I_1,I_2} = -\iint_0^1 P_{I_1,I_2}(i_1,i_2) \ln P_{I_1,I_2}(i_1,i_2) di_1 di_2$$
(10)

在灰度值只有 0-255 的图像系统中, 熵和联合熵都可以简化为离散的求和形式, 其中图像灰度值空间 的熵如式(10)。

$$H_I = \sum_{\mathbf{p}} h_I (I_{\mathbf{p}}) \tag{10}$$

其中 $h_I(i)$ 表示灰度值i对应的度值 $h_I(i)$ 的表达式如式(11)、(12)。

$$h_I(i) = -\frac{1}{n} \ln P_I(i) \tag{11}$$

$$P_{I}(i) = \frac{1}{n} \sum_{P} T[(i) = (I_{P})]$$
(12)

其中*P_l*(*i*)表示为灰度值*i*在图像*I*的出现频率,*n*表示为离散空间的大小,即在 0-255 的灰度空间中*n*为 256, T[*l*]是用于判断*l*是否成立的表达式,如果*l*成立则返回 1,否则返回 0。*h_l*(*i*)在计算机系统中可以生成 一个 1*256 的列表,*H_l*则是列表所有元素的和。

相似的联合熵也有对应的离散表达式。

$$H_{I_1I_2} = \sum_{\mathbf{p}} h_{I_1I_2}(I_{1\mathbf{p}}, I_{2\mathbf{p}})$$
(13)

$$h_{I_1,I_2}(i,k) = -\frac{1}{n} \ln \left(P_{I_1,I_2}(i,k) \right) \tag{14}$$

$$P_{I_1,I_2}(i,k) = \frac{1}{n} \sum_{P} T[(i,k) = (I_{1P}, I_{2P})]$$
(15)

在计算机系统中h_{I1,I2}(i,k)以一个 256*256 的二维图存放。

使用互信息作为代价,使代价最小化就可以使两幅图向更加相似的方向变化,用于计算互信息的两幅 图不是双目的左右图,而是一张基准图b和一张匹配图m,基准图可以任意选择双目的左图和右图,匹配图 是双目的另一张图经过视差图修正后的图像,图像I使用视差图D修正的操作定义为f_D(I)。

经过理想的视差图修正后的基准图和匹配图会高度相似,所以在联合概率密度图中,大部分点会落在 $I_1 = I_2$ 的直线上,因为在匹配过程中遮拦点无法正确匹配,会有部分点落在 $I_1 = I_2$ 的直线外,遮拦点因为 出现的概率较小,所以体现在 $P_{I_1,I_2}(i,k)$ 的图像中就是噪声点,在联合概率图 $P_{I_1,I_2}(i,k)$ 获取后和图 $h_{I_1,I_2}(i,k)$ 获取后都进行一次高斯滤波就可以大幅地减少。同样的熵的计算过程中也可以加入高斯滤波,这样可以减 少图像噪声对匹配的影响,使得视差图更加平滑完整。

加入高斯滤波的熵和联合熵的公式如下。

$$h_{I_1,I_2}(i,k) = -\frac{1}{n} \ln \left(P_{I_1,I_2}(i,k) * g(i,k) \right) * g(i,k)$$
(16)

$$h_{I}(i) = -\frac{1}{n} \ln(P_{I}(i) * g(i)) * g(i)$$
(17)

*g(i,k)表示对二维图进行高斯滤波,在本文中选择窗口大小为5*5, θ 为1.05651373的高斯核,*g(i)表示对一维队列进行高斯滤波,滤波器宽度选择5, θ 为1。计算联合熵 $h_{1,l_2}(i,k)$ 的步骤如图2所示。



3.1 代价函数

通过计算获取基准图和匹配图的 $h_I(i)$ 和 $h_{I_1,I_2}(i,k)$,可以获取两个列表和一张图用于查询在基准图b中点 p和经过视差图D修正后的匹配图m中点q的灰度值 I_{b_n} , I_{m_q} ,所对应的代价 $mi_{I_b,f_D}(I_m)(I_{b_n},I_{m_q})$ 。

$$mi_{I_{b},f_{D}(I_{m})}\left(I_{b_{\mathbf{p}}},I_{m_{\mathbf{q}}}\right) = h_{I_{b}}\left(I_{b_{\mathbf{p}}}\right) + h_{f_{D}(I_{m})}\left(I_{m_{\mathbf{q}}}\right) - h_{I_{b},f_{D}(I_{m})}\left(I_{b_{\mathbf{p}}},I_{m_{\mathbf{q}}}\right)$$
(18)

点**p**与点**q**在水平方向的距离为*d*,即b中点**p**的坐标为($\mathbf{p}_x, \mathbf{p}_y$),对应的m中点**q**的坐标为($\mathbf{q}_x, \mathbf{q}_y$)=($\mathbf{p}_x - d, \mathbf{p}_y$),视差修正的表达式定义如式(19)。

$$\mathbf{q} = e_{bm}(\mathbf{p}, d) \tag{19}$$

使用优化方法时习惯将代价函数最小化,互信息值越大表示匹配程度越高,所以单个像素 p 对于视差 d 的代价函数 $C_{MI}(\mathbf{p}, d)$ 需要加入一个负号。

$$C_{MI}(\mathbf{p}, d) = -mi_{I_b, f_D(I_m)}(I_{b_p}, I_{m_q})$$
(20)

现在就有了图像中每个点对于不同视差值的代价函数,整体的匹配代价就是所有像素点代价的和。如果逐 像素点地优化每个像素的代价函数使其最小化,将无法考虑邻域像素点间的约束,因为图像会受到光照和 噪声的影响,左右图对同一位置的值可能不同,获得的视差图可能难以连续稠密,甚至产生大量误匹配的 情况,所以需要将图像中像素间的约束加入到代价计算的环节,引入代价聚合环节。

3.2 代价聚合

如果不考虑像素间的约束,整体的代价就等于每个像素代价值的简单叠加,即下式。

$$E(D) = \sum_{\mathbf{p}} \left(C(\mathbf{p}, D_{\mathbf{p}}) \right)$$
(21)

*D*_p表示视差图在p点处的视差值。在实际的三维空间中,位置不同特别是深度有明显区别的物体的边缘才会产生明显的深度变化,在物体的表面,一般不会有深度剧烈变化的地方,表现在视差图中就是理想的视差图在整个物体表面要连续稠密,在深度明显变化的地方边界显著。在具有该先验条件的情况下,就可以向代价函数加入正则化项予以约束。

在此使用一种常用的使图像连续光滑的正则化方法,即对容易造成图像不连续不稠密的情况予以惩罚。 基本想法是如果某点的邻域点的视差值与该点的视差值的差为 1,即视差图中存在相邻两点值的差为 1, 则进行惩罚,因为邻域点产生比较小的视差的差值,会让图像产生椒盐噪声一样的噪点,不符合对视差图 的先验,所以给予代价函数惩罚,这样可以使得视差图稠密,惩罚系数P₁;而某点邻域的视差值的差值大 于 1 也会进行惩罚,惩罚系数设为P₂,惩罚系数P₂应该不小于P₁,这一项是为了保证视差图的平滑,当视 差发生剧烈跳变时则会有较大的惩罚,这一项也是关于视差图光滑的先验,保证**p**点确实到达了物体的边 缘,与邻域点会有足够大的视差值的变化,用来抵消P₂的惩罚,所以P₂也不适合过大,否则难以保持边缘。

加入了正则化项的代价聚合函数如下。

$$E(D) = E_{data} + E_{smooth}$$

$$E_{max} = \sum_{m}^{m} (C(\mathbf{p}, \mathbf{p}))$$
(22)

$$E_{data} = \sum_{\mathbf{p}} (C(\mathbf{p}, D_{\mathbf{p}}))$$
$$E_{smoot/i} = \sum_{\mathbf{q} \in N_{\mathbf{p}}} P_1 T[|D_{\mathbf{p}} - D_{\mathbf{q}}| = 1] + \sum_{\mathbf{q} \in N_{\mathbf{p}}} P_2 T[|D_{\mathbf{p}} - D_{\mathbf{q}}| > 1])$$
(23)

N_p表示点p的邻域,想要获得理想的视差图就需要优化代价聚合函数,使得代价最小化。

4 多方向的优化

E(*D*)对视差*d*是不可导的,所以无法使用梯度下降法和和高斯法进行优化,对于双目系统而言,视差*d*的取值一定为正,而且小于一个全局最大的视差值*d_{max}*,如果不确定一个*d*增长的方向,该优化问题则无法满足多项式时间,即 NP 完全问题,所以确定一个*d*增长的方向就可以暴力地计算某一点**p**对每个视差值*d*对应的代价*G*(**p**,*d*),取最小代价所对应的视差作为输出视差,该方法被称为线扫描优化。



在代价聚合函数中每个像素点的代价都和其邻域像素点有关联,而在像素级别的图像系统中邻域点包括上、下、左、右、左上、左下、右上、右下八个方向,所以在视差值d在增长过程中,即在线扫描优化的过程中,可以从多个方向进行优化。

从不同方向进行优化的结果会不同,仅仅从一个方向进行优化不能很好地利用邻域的约束,会在视差 图中产生大量条纹瑕疵,所以通常做法是将多个方向的代价函数叠加,再使用 WTA 原则,选取代价最小 对应的视差值作为输出视差值,下式是多方向代价函数叠加得到最终代价函数,**r**表示优化的方向,*G*_r(**p**,*d*) 表示在**r**方向上的点**p**对于视差*d*的代价函数。

$$S(\mathbf{p}, d) = \sum_{r} G_{r}(\mathbf{p}, d)$$
(24)

其中单方向的代价函数如下。

$$G_{r}(\mathbf{p},d) = C(\mathbf{p},d) + \min(G_{r}(\mathbf{p}-r,d),G_{r}(\mathbf{p}-r,d-1)+P_{1},$$

$$G_{r}(\mathbf{p}-r,d+1) + P_{1},\min_{i}G_{r}(\mathbf{p}-r,i)+P_{2}) - \min_{k}G_{r}(\mathbf{p}-r,k)$$
(25)

单方向的代价函数的表达式分为三个部分,第一项 $C(\mathbf{p},d)$ 表示点 \mathbf{p} 自身的代价值,第二项是一个最小化项,总共包括四项, $G_{\mathbf{r}}(\mathbf{p}-\mathbf{r},d)$ 表示路径相邻点的当前视差代价聚合值, $G_{\mathbf{r}}(\mathbf{p}-\mathbf{r},d-1) + P_1$ 和 $G_{\mathbf{r}}(\mathbf{p}-\mathbf{r},d+1) + P_1$ 表示路径相邻点的视差差值为1的代价聚合值+ P_1 , $min_iG_{\mathbf{r}}(\mathbf{p}-\mathbf{r},i) + P_2$ 表示路径相邻点的视差插值大于1的最小代价聚合值+ P_2 ,第三项 $min_kG_{\mathbf{r}}(\mathbf{p}-\mathbf{r},k)$ 也是一个最小化项,表示路径相邻点的视差插值大于1的最小代价聚合值。该公式实现了一种代价蔓延的过程,当前代价聚合值由当前代价和路径上一点的加了惩罚的最小代价聚合值所决定,即第一第二项,而第三项是为了在代价蔓延的过程中防止数值过大。

r可以选择四个方向,也可以选择八个方向甚至更多,方向越多,邻域点的约束越显著,但是计算成本 越高,本文的实验数据中,将使用上下左右四个方向的优化。

将多方向的代价函数叠加获得S(p,d)的函数,使用 WTA 原则最小化S(p,d)

$$D_{\mathbf{p}} = argmin_d S(\mathbf{p}, d) \tag{26}$$

 $D_{\mathbf{p}}$ 表示视差图中点**p**的视差值, $argmin_dS(\mathbf{p}, d)$ 表示从 $S(\mathbf{p}, d)$ 函数中选择使S最小的d值作为输出值 d_{out} , d的取值范围是从 0 到 d_{max} 的整数, d_{max} 称为视差的搜索范围。对所有点求解输出视差值就可以构 建视差图 D_{\circ}

5 迭代和输出

5.1 迭代优化

在第3和第4部分中实现了一种迭代优化的过程,在基于互信息的代价函数的计算过程的第一步就需要视差图D的输出,经过代价函数的构建和优化,又输出了视差图D,在此命名用于输入的视差图为D_{in},经过计算输出的视差图为D_{out},每次的迭代优化,视差图都会更加接近真实视差图,即经过视差图修正的基准图和匹配图的互信息将会提升,直到互信息到达允许到达的峰值。

$$\sum_{\mathbf{p}} m i_{I_b, f_{D_{out}}(I_m)} \left(I_{b_{\mathbf{p}}}, I_{m_{\mathbf{q}}} \right) \ge \sum_{\mathbf{p}} m i_{I_b, f_{D_{in}}(I_m)} \left(I_{b_{\mathbf{p}}}, I_{m_{\mathbf{q}}} \right)$$
(27)

在迭代的第一步没有可用的视差图,所以迭代第一步的输入视差图可以是任意一张图,通常的做法是 使用一张随机图作为迭代的初始条件,随机图可以保证迭代的第一步可以有效地使代价下降,使得后面的 迭代可以向期望的方向进行,程序迭代流程图如下。



图 6 程序迭代流程图

5.2 视差图输出

因为给定的初始条件不同和双目图像内容不同,所有每次迭代代价下降的趋势会有差异,但是通过实 验数据观测,五次迭代后输出的视差图就没有显著变化,互信息值趋于稳定,所以本文实验选择相对保守 的迭代次数,所有数据集都是迭代优化五次的结果。

6 实验数据

本文使用的双目图像数据集来自 http://vision.middlebury.edu/stereo/eval/,硬件选用 AMD Ryzen1600 处理器,两路 8G DDR4 2400Hz 内存,显卡为 GTX1050TI,软件平台为 Ubuntu18.04 操作系统,Cmake 编译器,OpencCV3.2 图像处理框架,未使用并行计算框架和算法。



图 7 本文实验结果的视差图



图 8 局部匹配算法 SAD 的视差图

使用 evkit 软件测试视差的数值,除了边缘误匹配点和无效匹配区域,大部分区域视差值的误差很小。 本文使用 2014年的数据集中的 Mask 和 Vintage 作对比实验,与局部匹配算法 SAD 和全局匹配算法 SPSS 进行比较,从图 7 和图 8 中观察,与 SAD 算法相比,本文算法表现出了全局匹配算法的优势, *E_{smoot}*,项 的加入让本文算法在低纹理区域也可以保持视差图的连续稠密,减少了对低纹理区域误匹配的情况,视差 图相对比较完整;图 7 和图 9 比较, SPSS 算法的结果更加完整,物体的轮廓更加清晰,但是相比于本文算 法,SPSS 算法舍弃了更多对细节的敏感,对于物体上有较小视差变化的部分进行了"平滑",本文算法在 参数调节环节也考虑了对小视差变化敏感度的取舍,调节参数*P*₁和*P*₂的数值就是调节算法对于视差图的先 验,从图 10(a)(b)的比较中观测,增大*P*₂的数值可以让视差图更加完整连续,无效匹配区域也有所减少,但 是对于物体表面有较小深度变化的区域,*P*₂较大的视差图没法很好的表现,在算法投入到特定场景时,可 以根据需求调节参数*P*₁和*P*₂的大小来调整视差图输出。



 (a)Mask
 (b)Vintage

 图 9 全局匹配算法 SPSS 的视差图



图 10 本文算法使用不同P₂输出的视差图

边缘的误匹配点的原因有很多,有可能来自于图像的噪声、算法的正则化项、代价模型等,在保持细节和保持完整稠密中本文算法选择了偏向后者。本文的视差图都是左视差图,所以视差图的左边区域会有 大面积无效区域,属于正常情况。

7 结束语

通过数据观测,该方法获取的视差图在整体上保持了连续稠密的先验,这得益于代价聚合函数中设置的正则化项以及P₁P₂的惩罚系数的调整,但是对正则化项保持图像平滑和保持边缘的同时,也牺牲了图像对于深度有细微变化区域的敏感程度,表现在视差图中就是物体表面的细节无法很好地捕捉。

对于低纹理区域也能有比较好的效果,基本不会有局部匹配算法中低纹理区域匹配出现大面积空洞的 情况,这是全局匹配对于局部匹配最明显的优势,将全局点都带入代价计算环节,在代价聚合环节加入邻 域点的约束,让匹配算法可以像素点视差的计算考虑到图像整体的内容,除了邻域点的约束外 Qingxiong Yang 还提出一种多尺度间的约束方法[®],但是全局匹配的代价也是巨大的,参与代价计算的点会非常多, 而且计算量会随着图像尺寸指数型增长,计算步骤也比较复杂,计算成本比较高,所以如果需要投入到实 时的应用场景中,计算平台的优化和算法底层的优化就会十分重要。

单一的左图作为基准图右图作为匹配图的策略,对于右图中的遮拦点无法很好地估计视差,会出现错

误匹配值,甚至是没有匹配结果,所以将左右图对应基准图和匹配图的策略调整,生成左、右视差图,然 后通过视差图后处理^[11],对遮拦区域的点进行估计,得到完整的视差图,这也是立体匹配算法领域中常用 的手段,对于不同的匹配算法可以选择合适的后处理的方法,用来解决不同匹配算法的缺陷。

参考文献:

- [1] Kaehler Adrian, Bradski Gary. Learning OpenCV3[M]. O Reilly Media, 2016. 704-710.
- [2] Zhang Kang, Fang Yuqiang, Min Dongbo, et al. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)[C]. 2014. 1590–1597.
- [3] Boykov Y, Veksler O, Zabih R. Fast Approximate Energy Minimization via Graph Cuts[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001, 23(11).
- [4] Kolmogorov V, Zabih R. Computing Visual Correspondence with Occlusions using Graph Cuts[J]. IEEE International Conference on Computer Vision, 2001, I: 508–515.
- [5] Sun J, Zheng N-N, Shum H-Y. Stereo Matching Using Belief Propagation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(7).
- [6] Sun J, Li Y, Kang S B, et al. Symmetric Stereo Matching for Occlusion Handling[J]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, II: 399–406.
- [7] Yang Qingxiong, Yang Ruigang. Stereo Matching with Color-Weighted Correlation[J]. Hierarchical Belief Propagation and Occlusion Handling, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(3).
- [8] Yang Qingxiong. A non-local cost aggregation method for stereo matching[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2012. 26.
- [9] Yang Qingxiong. Stereo Matching Using Tree Filtering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(4).
- [10] Bishop Christopher M. pattern recognition and machine learning[M]. London: Springer, 2011. 39-45.
- [11] Egnal Geoffrey, Wildes Richard P. Detecting Binocular Half-Occlusions: Empirical Comparisons of Five Approaches[J]. Pattern Analysis and Machine Intelligence, 2002, 24(8): 1127–1133.
- [12] 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, (02).
- [13] 周文晖,林丽莉,顾伟康. 一种鲁棒的基于互信息的实时立体匹配算法[J]. 传感技术学报, 2006, (04).
- [14] 李金凤. 立体匹配算法的研究[J]. 黑龙江科技信息, 2015, (27).
- [15] 罗大思,王进华. 基于双目视觉的立体匹配算法研究[J]. 微型机与应用, 2016, (20).

Stereo Matching Method based on Mutual Information

DU Si-ao, YIN Ye-an, WU Wen-jun

(School of Mathematics and Computer Science, Wuhan Textile University, Wuhan Hubei 430200, China)

Abstract: This paper introduces a stereo matching method based on grayscale mutual information, which is used to obtain disparity map from binocular map. In this method, the gray-scale mutual information is used as an index to measure the similarity of the input image, and the cost of the single-pixel point is constructed. Then the constraint of the neighborhood point is added in the cost aggregation, and finally the multi-directional sweep line optimization is used to optimize the cost aggregate function. Repeating the optimization process is to make the output disparity map gradually close to the real situation. This method uses a random graph as the initial state of the iteration, iterates a fixed number of times and output. Gray-scale mutual information as a cost criterion makes the method faster in the calculation of the cost than the general global matching method, and has certain robustness to illumination. The constraint of the domain point makes the disparity map more smooth and dense.

Key words: stereo matching; mutual information; cost; optimization