

一种基于 Walsh 变换的立体匹配单元

夏永泉, 杨静宇

(南京理工大学计算机与科学系, 南京 210094)

摘要: 提出了一种以 Walsh 变换系数作为匹配基元的双目立体匹配方法。通过 Walsh 正交变换核图像, 变换得到 Walsh 系数, 图像的特征被嵌入到该系数中, 费用函数采用了 Walsh 变换系数作为匹配基元, 而不是采用传统的像素灰度值。该文给出了算法的费用函数、匹配过程中相关的数据和最后的视差图。试验结果表明, 将 Walsh 系数作为一种匹配基元的方法是可行和有效的。

关键词: 立体匹配; Walsh 变换; 变换核; 系数; 视差计算

A Stereo Matching Approach Based Walsh Transform

XIA Yongquan, YANG Jingyu

(Department of Computer Science & Technology, Nanjing University of Science and Technology, Nanjing 210094)

【Abstract】This paper proposes a new binocular stereo matching approach based on Walsh transformation. This approach gets the Walsh coefficient through transform image by using Walsh orthogonal transform kernel, after the transformation, the coefficient embeds the characters of images, so the characters of image are represented in the form of Walsh coefficients. The provided algorithm in the paper uses the Walsh transform coefficient as match base to replace the pixel gray value. The result disparity, relevant data and figures are listed in the paper and the result indicates that this approach of using Walsh coefficients as match primitives is possible and valid.

【Key words】 Stereo matching; Walsh transform; Transform kernel; Coefficient; Disparity computation

为了有效地和快速地对图像进行处理和分析, 将定义在原图像空间上的图像以某种形式转换到另外一些空间, 并利用在这些空间的特有性质, 方便地对图像进行处理和加工, 最后转换到图像空间以达到所需要的效果, 如基于可分离变换的傅利叶变换、快速傅利叶变换以及Walsh变换和基于统计变换的霍特林变换(KL变换)^[1]等。Walsh变换是一种可分离的正交变换, 经过变换能将图像的特征尺度转化到Walsh系数中, 文中正是据此提出了一种采用Walsh变换系数作为匹配基元的双目立体匹配方法, 本方法利用Walsh变换系数的均值作为图像的匹配基元, 而不是采用传统的像素灰度值作为匹配基元。

1 Walsh 变换

Walsh 变换是可分离的正交变换, 在图像处理领域得到了广泛的应用, 下面简单介绍一般的正交变换和 Walsh 正交变换, 详细的资料可见文献[2~5]。

1.1 Walsh 正交变换

设 $f(x,y)$ 2-D 下的离散图像, 大小为 $M \times N$, 则一般的 2-D 正交变换和逆变换为

$$W(u,v) = \frac{1}{N^2} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-1-i}(u) + b_i(y)b_{n-1-i}(v)} \quad (1)$$

$$f(x,y) = \frac{1}{N^2} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} W(u,v) \prod_{i=0}^{n-1} (-1)^{b_i(x)b_{n-1-i}(u) + b_i(y)b_{n-1-i}(v)} \quad (2)$$

其中, $b_i(x)$ 为 x 的二进制表达式的第 i 位。

$$k_f(x,u) = \frac{1}{N} \prod_{i=0}^{n-1} (-1)^{b_i(x) \cdot b_{n-1-i}(u)}$$

$$k_f(y,v) = \frac{1}{N} \prod_{i=0}^{n-1} (-1)^{b_i(y) \cdot b_{n-1-i}(v)}$$

则式(1)和式(2)可改写为如下的形式:

$$W(u,v) = k_f(x,u) \cdot f(x,y) \cdot k_f(y,v)^T \quad (3)$$

$$f(x,y) = k_f^{-1}(x,u) \cdot W(u,v) \cdot k_f^{-1}(y,v)^T \quad (4)$$

$$\mu, x = 0, 1 \dots, M-1, v, y = 0, 1 \dots, N-1$$

$k_f(x,u)$ 和 $k_f(y,v)$ 为变换核, 定义为

$$k_f(x,u) = k_f(y,v) = WAL(i,j) = \prod_{q=0}^{n-1} (-1)^{b_q(i)b_{n-q-1}(j)} \quad (5)$$

$$k_f(x,u) = k_f(x,u)^T = k_f^{-1}(x,u) = k_f^{-1}(x,u)^T \quad (6)$$

满足式(5)和式(6)的式(3)和式(4)即为 Walsh 正交变换和逆变换。本文算法采用了 8×8 的正交变换核, 如表 1 所示(略去 $1/N$, 变换核的值只有 $+1$ 和 -1)。

表 1 Walsh 8×8 变换核

j \ i	0	1	2	3	4	5	6	7
0	+1	+1	+1	+1	+1	+1	+1	+1
1	+1	+1	+1	+1	-1	-1	-1	-1
2	+1	+1	-1	-1	+1	+1	-1	-1
3	+1	+1	-1	-1	-1	-1	+1	+1
4	+1	-1	+1	-1	+1	-1	+1	-1
5	+1	-1	+1	-1	-1	+1	-1	+1
6	+1	-1	-1	+1	+1	-1	-1	+1
7	+1	-1	-1	+1	-1	+1	+1	-1

由表 1 可知, Walsh 正交变换核满足下列正交条件:

(1)任意给定 r_i 和 r_j

$$\sum_{c=0}^{n-1} W(c, r_i) \cdot W(c, r_j)^T = \begin{cases} 1 & \text{if } r_i = r_j \\ 0 & \text{if } r_i \neq r_j \end{cases}$$

(2)任意给定 c_i 和 c_j

$$\sum_{r=0}^{n-1} W(c_i, r) \cdot W(c_j, r)^T = \begin{cases} 1 & \text{if } c_i = c_j \\ 0 & \text{if } c_i \neq c_j \end{cases}$$

Walsh 正交变换核可以应用到以下不同的情况:

(1)两维:

作者简介: 夏永泉(1972—), 男, 博士生, 主研方向: 计算机视觉, 模式识别, 人工智能; 杨静宇, 教授、博导

收稿日期: 2005-09-23 **E-mail:** xyqmouse@163.com

$$W(u,v) = k_f(x,u) \cdot f(x,y) \cdot k_f(x,u) \quad (7)$$

(2)一维：

$$W(u,y) = k_f(x,u) \cdot f(x,y) \quad (8)$$

(3)图像的一行：

$$W_r(u) = k_f(x,u) \cdot (\text{row}_i)^T \quad (9)$$

1.2 Walsh 系数

文中算法采用了式(8)计算每行的Walsh系数,当 $N=2^n$ 时,对 $M \times N$ 图像,定义如下的Walsh系数矩阵:

$$L_k(\xi) = [a_{k0}(\xi), a_{k1}(\xi) \cdots, a_{k(n-1)}(\xi)]^T \quad (10)$$

$$R_k(\xi) = [b_{k0}(\xi), b_{k1}(\xi) \cdots, b_{k(n-1)}(\xi)]^T \quad (11)$$

$L_k(\xi)$ 和 $R_k(\xi)$ 分别代表左右图像第 k 行位置 ξ 处的特征矩阵。

$$a_i(\xi) = \sum_{j=0}^{n-1} WAL(i,j) \cdot x_{i(j+\xi)} \quad \xi = 0, 1, \dots, N-n$$

$x_{ij} = f(i, j)$ 为 i 行 j 列像素的灰度值。

$b_i(\xi)$ 有类似的定义。本算法中采用了 8×8 的Walsh变换核,即 $n=8$ 。对Tsukuba图像对(如图1所示)的100行, $a_i(\xi)$ 和 $b_i(\xi)$ 如图2所示。

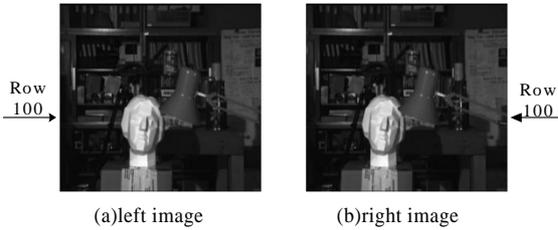


图1 实验用的立体图像对

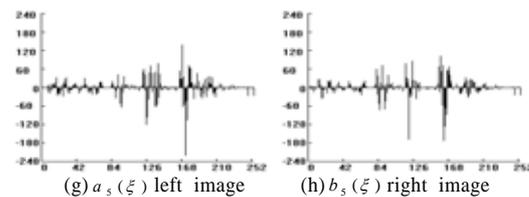
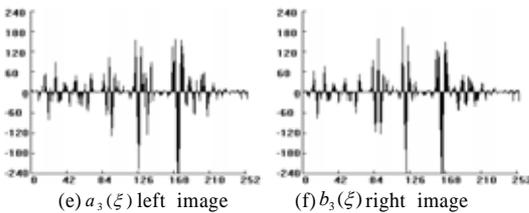
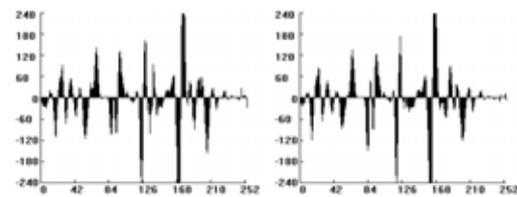
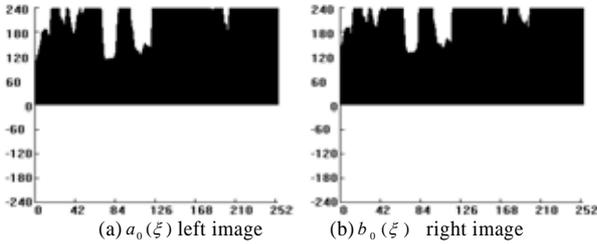


图2 图像对100行Walsh系数

Walsh系数是线性、独立两两正交,因此匹配系数的结果是两两独立的;系数的计算简单,只是加法和减法操作;按照Walsh基函数,图像的特征以非常精确的方式被尺度化到Walsh系数中,但在实际应用中应注意高阶系数产生比较小的幅度,如果单独采用一阶的系数作为特征,则应该采用角小阶的系数。本算法采用了1~7阶的系数均值作为特征,即对每阶采用相同的权值,而忽略了0阶Walsh系数,因为它只是像素灰度值的简单相加。

2 匹配过程

匹配过程分为两步:(1)计算左右图像的Walsh系数,确定匹配基元;(2)确定费用函数,然后计算得到最后的视差。

2.1 匹配基元的选择

从图2、图3中可以看出,随着阶数的增大,Walsh系数特征幅度减小, a_0 就是窗口 n 内的像素灰度值的简单相加,不适合作为匹配基元,可以选择1~7阶任何一阶系数特征作为匹配基元,本文算法选择了1~7阶Walsh系数之和的均值 $\bar{m}(x, \xi, n)$ 作为匹配基元,定义如下:

$$\bar{m}(x, \xi, n) = \frac{1}{n-1} \sum_{i=1}^{n-1} x_i(\xi)$$

那么式(10)和式(11)可重写为

$$\bar{L}_k(\xi) = [\bar{m}(a, \xi, n)]$$

$$\bar{R}_k(\xi) = [\bar{m}(b, \xi, n)]$$

$\bar{L}_k(\xi)$ 和 $\bar{R}_k(\xi)$ 分别为左右图像第 k 行 ξ 处的特征, $0 \leq \xi \leq N-n$ 。例如左右图像行100的Walsh系数均值 $\bar{L}_{100}(\xi)$ 、 $\bar{R}_{100}(\xi)$,如图3所示。

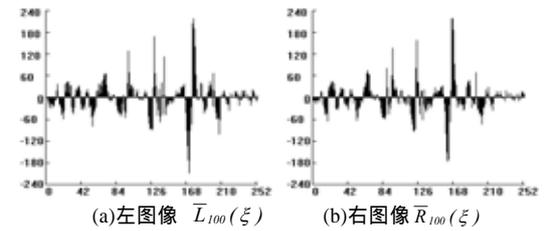


图3 行100系数(1~7阶)均值

2.2 费用函数

邻域相似性在许多算法在立体匹配算法中经常使用,文中算法利用Walsh系数1~7阶系数均值作为匹配特征,然后以本特征的邻域相似性作为匹配测度,本匹配是在上节中提到的已成功匹配的1阶Walsh系数零交叉点约束下进行。

左图像 k 行 ξ 处邻域和右图像 k 行 η 处邻域的Walsh系数相似性测度用下列的 $C_k(a, b, \xi, \eta, n, r)$ 来表示。

$$C_k(a, b, \xi, \eta, n, r) = \frac{1}{s(w)} \sum_{r \in w} \langle \bar{m}(a, \xi+r, n) - \bar{m}(b, \eta+r, n) \rangle \quad (12)$$

$$\xi \in [l-1, N-1-l], \quad \eta \in [l-1, N-1-l]$$

其中, l 为邻域半径, $s(w)$ 为邻域 w 的面积,如八邻域,那么 $s(w)=8$,算法中采用了八邻域。在对左图像第 k 行 ξ 点,在右图像中 ξ 点的匹配窗口 $[\xi - \text{Max}(d), \xi + \text{Max}(d)]$ 中寻找匹配点, $\text{Max}(d)$ 为允许的最大视差,本文的算法采用了下面的费用函数来进行匹配:

$$C(k, \xi, d_i) = C_k(a, b, \xi, \xi + d_i, n, r)$$

$$d_i = [-\text{Max}(d)/2, \text{Max}(d)/2], \quad \xi = 0, 1, \dots, N-1-\text{Max}(d)/2$$

可得到第 k 行 ξ 处的视差 $d_k(\xi)$ 如下:

$$d_k(\xi) = \langle d_i \mid \text{when}(\min(C(k, \xi, d_i))) \rangle$$

(下转第59页)