

基于窗口滤波与均值滤波的深度图像实时修复算法

李腾飞, 刘嘉敏, 段 勇

(沈阳工业大学信息科学与工程学院, 辽宁 沈阳 110870)

摘 要: 使用Kinect作为视频图像的输入设备, 获取的连续帧的深度图像会在边缘出现空洞像素点和像素点随机抖动的问题。本文针对Kinect获取的连续帧深度图像提出了加权窗口-除零均值滤波算法。先使用提出的加权窗口滤波算法对空洞像素填补, 然后采用改进后的除零均值滤波算法对图像进行平滑与去噪声处理。实验结果表明, 相比双边滤波、中值滤波、均值滤波和高斯滤波算法, 本文提出的算法能够有效的减少图像中的抖动像素, 去除深度图像中的噪声信息, 使图像边缘信息平滑, 并且保证视频输入有较高的帧率。

关键词: 深度图像; 视频; Kinect; 抖动; 空洞

中图分类号: TP391 **文献标识码:** A

A Depth Image Real-Time Image Inpainting Algorithm Based on Window Filtering and Mean Filtering

LI Tengfei, LIU Jiamin, DUAN Yong

(School of Information Science and Engineering, Shenyang University of Technology, Shenyang 110870)

Abstract: As an input device of video images, Kinect can obtain the depth images of successive frames. However, there exists the issue of edge holes and random jitter of pixels in the depth video images. In this paper, the weighted window filter and the non-zero mean filtering algorithm are proposed for the successive frame depth image acquired by Kinect. Firstly, the depth video image is repaired by the proposed weighted window filtering algorithm. The image is then smoothed and the noise is eliminated by using an improved non-zero mean filtering algorithm. The experimental results show that the algorithm can more effectively remove a large number of jitter pixels in video images and smooth edge information than the algorithms of bilateral filtering, median filtering, mean filtering and Gaussian filtering. The algorithm ensures a high frame rate of the video input.

Keywords: depth image; Kinect; filter; jitter; hole

1 引言(Introduction)

Kinect是微软公司推出的一款体感设备, 它强大功能, 价格低廉且开发方便, 它已被应用到计算机视觉的各个应用中, Nagori^[1]等人使用Kinect实现聋哑人手语翻译, Fankhauser^[2]等人利用Kinect实现了机器人导航系统, 以及Chen^[3]等人以Kinect为平台实现增强现实的游戏系统。

由于Kinect发射的红外信号被目标物体吸收, 以及其他物体遮挡返回的红外信号等原因, 会造成图像空洞及抖动^[4]。为此需要对具有像素空洞和抖动深度进行修复, Du^[5]等人采用了改进双边滤波算法和聚类算法, 较好地修复了图像中空

洞; 彭诚^[6]等人利用形态学滤波算法, 结合双边滤波算法应用到深度图像处理当中; Hu^[7]等人提出了利用彩色图像引导局部正则化表示的方法来改进双边滤波算法。上述工作虽然修复的效果良好, 但没考虑连续帧深度图像所产生的像素抖动, 而且效率低, 无法保证对连续帧深度图像的实时处理。

本文提出了一种平衡了效率与处理效果的滤波算法, 即加权窗口滤波-除零均值滤波算法。首先使用加权窗口滤波算法来填补空洞和抖动的像素部分, 然后使用除零均值滤波算法对图像行平滑处理。实验通过与主要滤波算法进行比较, 本文提出的算法在效率与图像结果均达到理想程度。

2 加权窗口-除零均值滤波算法(Weighted window filter and non-zero mean filter algorithm)

2.1 加权窗口滤波算法

Kinect获取的每帧图像存在着空洞像素及抖动像素, 这影响后续的图像分割处理, 为此有必要消除每帧图像中的空洞像素及抖动像素。

首先通过Kinect获取每帧原始深度图像信息, 将原始深度图像信息由16位灰度图像转换为8位灰度图像, 以便后续处理。

设转换后的8位灰度图像 $I(x,y)$ 为在深度图中 (x,y) 处像素点的灰度值。如果有 $I(x,y)=0$, 则该点为空洞点。在图像中, 以该像素点为中心, 建立一个大小为 5×5 的正方形窗口, 遍历窗口中的所有非0灰度值的像素, 并将每个灰度值出现的次数使用哈希表记录下来, 如式(1)所示。

$$\begin{aligned} 0 \leq \text{Hash}[I(x,y)] &\leq 24 \\ 1 \leq I(x,y) &\leq 255 \end{aligned} \quad (1)$$

由于设定以某灰度值为0的像素点为中心, 遍历其周围 5×5 大小的正方形区域的像素, 所以每个像素的灰度值最多会出现24次。深度图像为8位灰度图像, 每个像素的灰度值范围即为0到255, 由于灰度值为0的像素不在计算范围之内, 所以范围限定为1到255。

在该算法中设定两个阈值, 分别为threshold1与threshold2, 判断如果在窗口中在边界上的非0像素个数是否大于threshold1且非边界上的非0像素值是否大于threshold2, 那么计算在窗口中像素的加权平均值, 如式(2)所示。

$$I_{\text{new}}(x,y) = \frac{\sum_{i=1}^{255} H_i \times i}{\sum_{i=1}^{255} H_i} \quad (2)$$

$I_{\text{new}}(x,y)$ 表示更新后的像素值, H_i 为像素值 i 出现的频率。

如果不满足阈值的条件, 则 $I_{\text{new}}(x,y)=0$ 。

例如, 在图1的 5×5 窗口中, 边界上的非0像素个数为12个, 非边界上的非0像素为6。

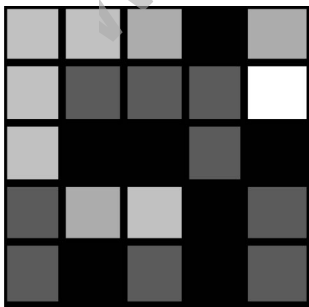


图1 5×5 窗口中的像素

Fig.1 Pixels in a 5×5 window

2.2 除零均值滤波算法

对深度图像的空洞像素及抖动像素修复后, 需要对图像

非空洞部分进行平滑处理, 以保证去除深度图像中出现的非空洞部分噪音。

除零均值滤波算法是在均值滤波^[8]算法基础上进行了改进, 传统的均值滤波算法是计算窗口中所有像素值的均值, 本文提出的改进算法不考虑灰度0值的空洞像素, 在计算时直接将它们剔除, 其值计算采用式(3)。

$$\text{pixel}_{(i,j)} = \frac{1}{k} \sum_{x=x+1}^{x+4} \sum_{y=y+1}^{y+4} I(x,y) \quad (3)$$

式(3)中 $\text{pixel}_{(i,j)}$ 表示在图像中第 i 处像素经过除零均值滤波算法处理后的值, k 表示 5×5 窗口中非零像素的个数。

为了提高算法的效率, 在计算窗口中的像素均值时, 使用了积分图像的方法^[9]。建立一个与图像尺寸大小相同的二维数组 $S(i,j)$, 记录图像从第 $(1,1)$ 点的像素值到第 (i,j) 点像素值的累加和, 如式(4)所示。

$$S(i,j) = \sum_{x=1}^i \sum_{y=1}^j I(x,y) \quad (4)$$

$S(i,j)$ 的值根据式(5)来计算。

$$S(i,j) = S(i-1,j) + S(i,j-1) - S(i-1,j-1) + I(i,j) \quad (5)$$

根据式(5)可知, $S(i,j)$ 在时间复杂度 $O(R \times C)$ 之内计算出来, 即对图像进行预处理即可得到, 其中 R 和 C 分别为图像的行数和列数。

由于在滤波的窗口中非0的像素个数事先并不确定, 按照式(5)可计算从 $(1,1)$ 点的像素值到第 (i,j) 点所包含的零值像素的个数, 并保存到零值积分表 $Z(i,j)$ 中, $Z(i,j)$ 表示从 $(1,1)$ 点的像素值到第 (i,j) 点之间的0的个数, 采用式(6)计算其值。

$$\begin{cases} Z(i,j) = Z(i-1,j) + Z(i,j-1) - Z(i-1,j-1) + 1, & \text{当 } I(i,j) = 0 \\ Z(i,j) = Z(i-1,j) + Z(i,j-1) - Z(i-1,j-1), & \text{当 } I(i,j) \neq 0 \end{cases} \quad (6)$$

计算均值需要枚举图像中每一个非0像素点, 以及以该像素点为中心的窗口 W 中像素, 则计算时间复杂度为 $O(R \times C \times WL \times WL)$, R 、 C 、 WL 分别为图像的长、宽、窗口边长, 如图2所示。使用积分图像和零值积分表后, 计算窗口 W 中像素和, 以及0值的个数, 其时间复杂度降为 $O(1)$ 。

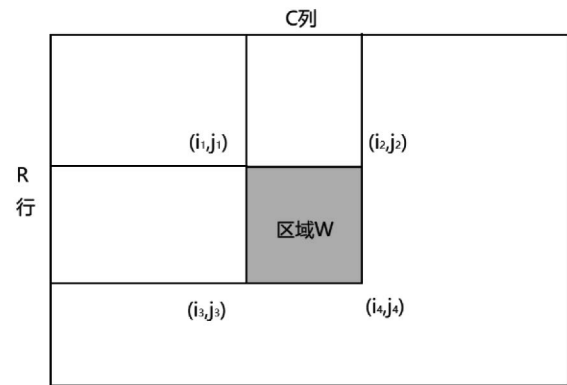


图2 计算 W 窗口像素和

Fig.2 The sum of the pixels value in window W 积分图像窗口 W 中的像素和用 SUM_w 表示, 其计算方法如

式(7)所示。

$$SUM_w = S(i_4, j_4) - S(i_2, j_2) - S(i_3, j_3) + S(i_1, j_1) \quad (7)$$

零值积分表窗口W中的零值个数用 $ZERO_w$ 表示, 计算方法如式(8)所示。

$$ZERO_w = Z(i_4, j_4) - Z(i_2, j_2) - Z(i_3, j_3) + Z(i_1, j_1) \quad (8)$$

窗口W中的均值用 $MEAN_w$ 表示, 计算公式如式(9)。

$$MEAN_w = \frac{SUM_w}{(i_4 - i_3 + 1) \times (j_4 - j_2 + 1) - ZERO_w} \quad (9)$$

3 实验结果(Experimental result)

本算法在VS 2013平台用C++实现, PC机的性能参数为Intel(R) Xeon(R) CPU E5-1620 v3 3.50GHz处理器, 16G内存下实现, 视频输入设备使用Kinect v2。

本文算法通过两次滤波, 得到高质量的图像, 如图3和图4所示。图3(a)为只使用加权窗口滤波方法的效果图, 图3(b)为两次滤波的效果图, 即先使用加权窗口滤波, 然后使用除零均值滤波算法。图4展示了仅使用加权窗口滤波算法和两次滤波算法的细节部分。由图3和4可以看出, 经过除零均值滤波处理后深度图像中, 图像更加平滑, 有效的去除了图像中的噪声。

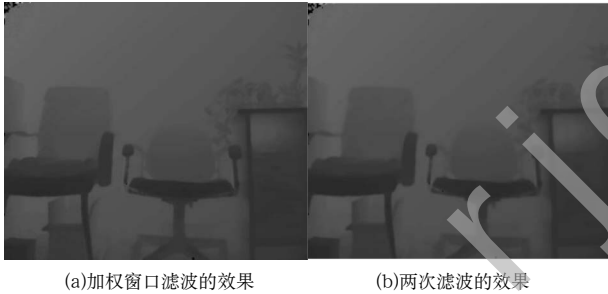


图3 本文算法获得的滤波结果

Fig.3 Result obtained by the algorithm in this paper

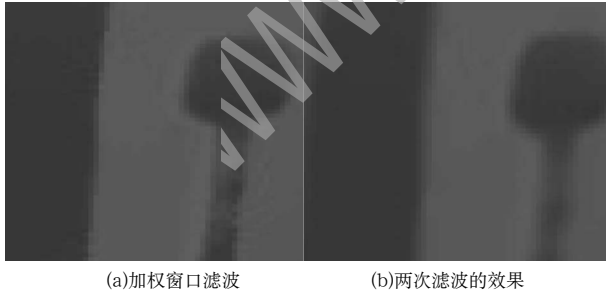


图4 图像局部细节的对比

Fig.4 Comparison of local details

为验证本文提出的算法的性能, 在计算效果和计算时间两方面与双边滤波算法、中值滤波算法、高斯滤波算法、均值滤波算法进行比较。

在测试中, 中值滤波参数窗口尺寸 $ksize=5$; 双边滤波参数像素邻域直径 $d=7$, 颜色空间参数 $\sigma C=131$, 坐标空

间参数 $\sigma S=100$; 均值滤波参数窗口尺寸 $ksize=(9, 9)$; 高斯滤波参数高斯内核大小 $ksize=(5, 5)$, x 方向标准差 $\sigma X=9$, y 方向标准差 $\sigma Y=9$ 。本文的加权窗口—除零均值滤波参数 $threshold1=2$, $threshold2=4$ 。5种算法滤波的效果如图5所示。

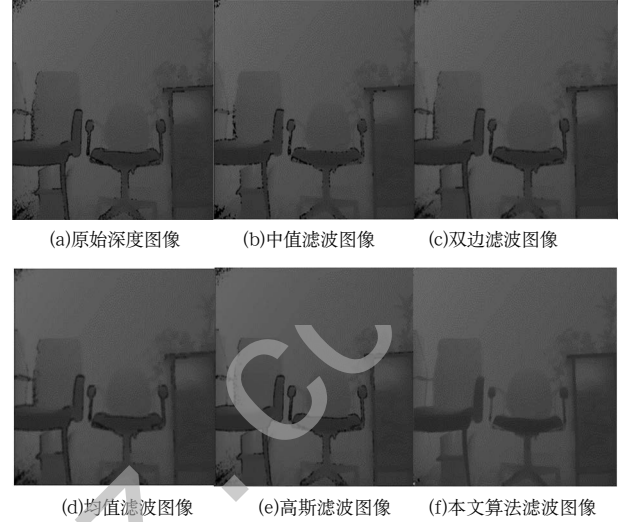


图5 原始图像及5种滤波算法的效果

Fig.5 The result of the original image and five filter algorithms

从图5可以看出, 相比其他四种滤波算法和原始深度图像, 图5(f)中的物体边缘空洞明和图像上部分物体的抖动像素明显减少, 且物体轮廓清晰。

原始图像与五种算法处理后灰度分布图如图6所示。图6中横轴为像素值, 范围在 $[0, 255]$; 纵轴为每个像素出现频率, 范围在 $[0, 20000]$ 。

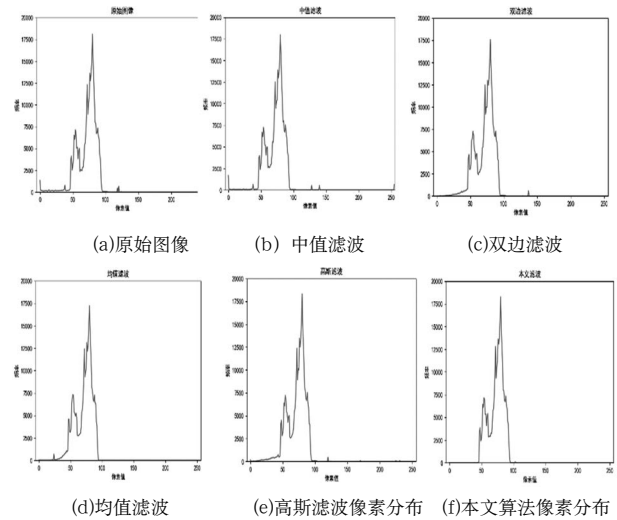


图6 原始图像及五种滤波算法的像素分布图

Fig.6 Grayscale distribution of the original image and five filter algorithms

由于在原始图像中存在大量抖动像素和空洞像素, 它们

主要集中在像素值为0附近部分，如图6(a)所示。各种滤波算法处理后的图像与原图像在像素分布中的峰值越接近，表示图像的细节信息越完整。由图6(f)看出，本算法不仅保留了图像当中灰度分布信息，而且有效的去除了空洞和抖动部分。由此可知，相比其他四种滤波方法，本文算法处理后得到的图像效果最佳。

对比滤波后效果，图7(a)与图7(b)经过二值化和膨胀^[10]处理后的帧差图。

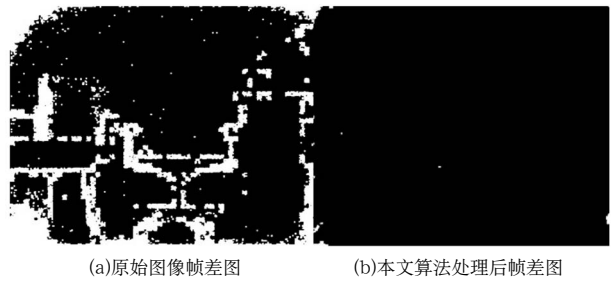


图7 原始图像与本算法处理后的静止场景帧差图

Fig.7 Static scene frame difference map of original image and image processed by the algorithm

图7(a)中静止场景相邻两帧的之间的抖动像素十分明显，在经过本文算法处理后，静止场景相邻两帧有零星的像素抖动，效果明显好于原始图像。

为了验证算法的效率，五种算法分别对连续70帧的同一幅深度图像进行滤波，其平均滤波时间如表1所示。

表1 五种滤波算法的运算时间

Tab.1 Operation time of five filtering algorithms

中值滤波	双边滤波	均值滤波	高斯滤波	本文滤波
0.053s	0.52s	0.081s	0.098s	0.349s

由表1看出，双边滤波算法运行时间最长，中值滤波运行时间最少，本文提出的加权窗口滤波算法运行时间略小于双边滤波。尽管本文算法平均运行时间较高于中值、均值和高斯滤波算法，但滤波效果好于它们，而且满足了视频图像实时处理的要求。

4 结论(Conclusion)

本文通过对实时Kinect深度图像特点进行分析，提出了加权窗口-除零均值滤波算法，对深度图像中的空洞像素进行修复；对均值滤波算法进行改进后，得到除零均值滤波对深度图像进行平滑处理。通过定性分析和定量实验结果表明，本算法在处理连续帧的深度视频图像上时间复杂度低，处理效率高，对深度图像的空洞像素及抖动像素有良好的处理效果，为下一步图像语义分割、图像标注、三维重建及三维物

体识别等处理提供良好的基础和平台。

本文算法仅使用了深度图像本身来进行实时性的图像修复工作，如何利用好Kinect中提供的其他数据信息来优化和提高深度图像质量是未来的研究方向。

参考文献(References)

[1] Nagori N P,Malode V.Communication Interface for Deaf-Mute People using Microsoft Kinect[C].International Conference on Automatic Control and Dynamic Optimization Techniques. IEEE,2017:640-644.

[2] Fankhauser P,Bloesch M,Rodriguez D,et al.Kinect v2 for mobile robot navigation:Evaluation and modeling[C]. International Conference on Advanced Robotics. IEEE,2015:388-394.

[3] Chen J Y,Liu C H,Hsieh C H,et al.Kinect augmented reality gear game design[C].International Conference on Applied System Innovation.IEEE,2017:373-375.

[4] 赵旭.Kinect深度图像修复技术研究[D].大连理工大学,2013:11.

[5] Du H,Miao Z.Kinect depth maps preprocessing based on RGB-D data clustering and bilateral filtering[C].Chinese Automation Congress.IEEE,2016:732-736.

[6] 彭诚,孙新柱.一种改进的深度图像修复算法研究[J].重庆工商大学学报(自然科学版),2016,33(1):65-69.

[7] Hu J,Hu R,Wang Z,et al.Color image guided locality regularized representation for Kinect depth holes filling[C]. Visual Communications and Image Processing.IEEE,2014:1-6.

[8] 宋清昆,马丽,曹建坤,等.基于小波变换和均值滤波的图像去噪[J].黑龙江大学自然科学学报,2016,33(4):555-560.

[9] 范帆.基于积分图像和Camshift的运动目标跟踪研究[D].浙江工业大学,2017:33-34.

[10] 安静.基于数学形态学的图像增强算法及其应用[D].西北师范大学,2016:7-8.

作者简介:

李腾飞(1993-),男,硕士生.研究领域:计算机图形学与虚拟现实.

刘嘉敏(1964-),女,博士,教授.研究领域:计算机图形学与虚拟现实、布局优化.

段勇(1978-),男,博士,教授.研究领域:多智能系统、机器学习.