

一种新的彩色圆盘角点检测方法

龚文彪 顾国华 钱惟贤 任建乐

南京理工大学 江苏省光谱成像与智能感知重点实验室, 江苏 南京 210094

摘要 圆盘靶标在摄像机标定及无人机侦察任务中目标的定位方面有着广泛的应用。针对目前图像角点检测的局限性,提出了一种自动提取红蓝彩色圆盘靶目标中心角点的检测方法。该方法利用HSV(Hue, saturation, value)颜色空间对红蓝彩色圆盘进行颜色分离,获得具有明显对称性和对比度强烈的灰度图像。然后运用提出的新型对称算子计算每个像素点的响应值作为角点的判断依据,对求得的角点在其邻域内抑制红蓝像素个数差来确保角点的对称性。最后通过Matlab软件上的仿真和现场可编程门阵列(FPGA)硬件上的实现,证明了该方法易于硬件化实现,实时性强,抗噪性能好,对图像旋转、伸缩和尺度变换都有良好的稳健性。

关键词 图像处理;彩色圆盘;角点检测;HSV颜色空间;对称算子

中图分类号 TN29 文献标识码 A doi: 10.3788/LOP51.031001

A New Algorithm for Color Disc Corner Detection

Gong Wenbiao Gu Guohua Qian Weixian Ren Jianle

Jiangsu Key Laboratory of Spectral Imaging and Intelligent Sense, Nanjing University of Science and Technology, Nanjing, Jiangsu 210094, China

Abstract Target disc images are widely used in camera calibration and target location for unmanned (UAV) scout task, so a method of automatically extracting feature point coordinates is proposed for the color on the two-dimensional (2D) center of the disc to solve the problems of present corner detection algorithms. The operation of color separation for the red and blue discs is completed with the method of hue-saturation-value (HSV) color space in order to get a gray image which has obvious symmetry and strong contrast. According to the corner points of the candidate red and blue pixels in the neighborhood with the distance less than a threshold number, the exact value of the corner is determined, and the response value of extracted large corner point is applied as a candidate by using a new type of symmetric operator. Experiment results of Matlab simulation and hardware implementation based on the field programmable gate array (FPGA) platform show that the proposed method is easy to implement in hardware technology, real-time and has good anti-noise ability. Also it has a good robustness to image rotation, stretching and scale transformation.

Key words image processing; color disc; corner detection; HSV color space; symmetric operators

OCIS codes 100.2000; 100.4999; 100.6890

1 引言

角点是在二维图像中 X 、 Y 方向上梯度值明显变化或图像边缘上曲率最大的点,一个好的角点对图像的旋转、伸缩和尺度变换都具有良好的稳健性^[1]。不同类型的角点具有不同的特点,现研究阶段大多数角点检测的方法都是针对一般的角点^[2],对于通用的角点检测及相关的改进方法,都很难避免算法量大、容易出现

收稿日期: 2013-10-12; 收到修改稿日期: 2013-11-27; 网络出版日期: 2014-03-01

基金项目: 国家自然科学基金(61101119)、江苏省普通高校研究生科研创新基金(CXZZ12_0183)、江苏省自然科学基金(BK2011699)

作者简介: 龚文彪(1990—),男,硕士研究生,主要从事红外图像处理和目标搜索跟踪方面的研究。

E-mail: gongwenbiao2011@163.com

导师简介: 顾国华(1966—),男,研究员,博士生导师,主要从事光电成像理论与技术、图像识别处理及应用技术方面的研究。E-mail: gghnjust@163.com

本文电子版彩色效果请详见中国光学期刊网 www.opticsjournal.net

错判的缺点。而在一些特定场合下,如摄像机的标定^[3]、无人机侦察任务中目标的定位^[4]、计算机视觉检测^[5]等,都可用到一些固定的靶标来减少算法的复杂度,提高系统的可靠性。

红蓝彩色对称圆盘具有一些独特性质^[6],根据其对称性和色彩的独特性可以完全跳出经典角点检测算法的思路^[7-8]。本文通过选择红蓝彩色圆盘作为靶标^[9],提出了一种新的检测方法来检测彩色圆盘的中心角点。该方法不但计算量小、原理简单、实时性好、无需人工干预,而且对于图像的旋转、伸缩和尺度变化都具有良好的检测效果,同时在 HSV(Hue,saturation, value)^[10]色彩空间中进行颜色分离,可得到稳定的灰度图像,从而大幅度提高检测方法的抗噪性能。

首先分析了常用角点检测算法在检测图像角点方面的局限性,然后介绍了一种新的对称算子对彩色圆盘的中心角点进行检测,并和 Harris 角点检测算法^[1]进行了比较,给出了实验对比结果。最后将该算法移植到基于 ALTERA 公司的 Cyclone IV EP4C115F29C7N 的现场可编程门阵列(FPGA)硬件平台上,验证了该算法可硬件化操作,实时性好。

2 角点检测算法

2.1 颜色识别和算法分析

彩色圆盘由红蓝相间的扇形组成,圆盘中心点为角点,如图 1 所示。

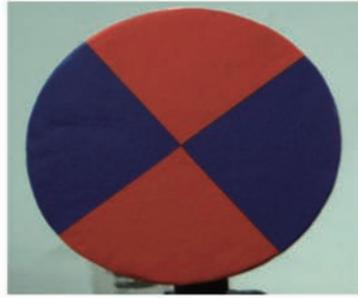


图 1 红蓝彩色圆盘

Fig.1 Red and blue color disc

利用彩色圆盘相对于灰度图像的优点是:进行颜色分割后得到的灰度图像抗噪性能好,且利用 HSV 彩色空间分离,不论亮度如何变化,同一颜色下其他相 h 值是不变的,提高系统的自适应能力。红绿蓝(RGB)到 HSV 彩色空间的转换按下式进行:

$$h = \begin{cases} 0^\circ, & \text{if } V_{\max} = V_{\min} \\ 60^\circ \times \frac{g-b}{V_{\max}-V_{\min}} + 0^\circ, & \text{if } V_{\max} = r \text{ and } g \geq b \\ 60^\circ \times \frac{g-b}{V_{\max}-V_{\min}} + 360^\circ, & \text{if } V_{\max} = r \text{ and } g < b \\ 60^\circ \times \frac{b-r}{V_{\max}-V_{\min}} + 120^\circ, & \text{if } V_{\max} = g \\ 60^\circ \times \frac{r-g}{V_{\max}-V_{\min}} + 240^\circ, & \text{if } V_{\max} = b \end{cases}, \quad (1)$$

$$s = \begin{cases} 0, & \text{if } V_{\max} = 0 \\ 1 - \frac{V_{\min}}{V_{\max}}, & \text{otherwise} \end{cases}, \quad (2)$$

$$v = V_{\max}, \quad (3)$$

式中 h, s, v 分别为 HSV 空间的色相、饱和度和亮度, r, g, b 为 RGB 空间中的三原色,分别对应于红色、绿色、蓝色; V_{\max} 为 (r, g, b) 中的最大值, V_{\min} 为 (r, g, b) 中的最小值。

由(1)~(3)式可知,对于不同的颜色,其 h 值是不同的,与其亮度值无关,因此对于红蓝色彩的检测,红色 h 的值为 0° ,蓝色 h 的值为 240° ,但为了保证算法在一定色彩范围内的自适应性,红色部分取 $h \in [350^\circ, 360^\circ) \cup [0^\circ, 10^\circ]$,蓝色部分取 $h \in [230^\circ, 250^\circ]$ 。因此算法先将 RGB 图像转到 HSV 颜色空间中,得

到每个像素点的色相值 h 。根据红、蓝部分 h 取值范围的不同, 获得对称性明显和对比度强烈的二值化红蓝图像。再由本文提出的一种新型对称算子, 检测圆盘的中心作为图像的角点, 该中心角点有如下性质: 角点附近的像素点关于角点中心对称的像素点其色相值 h 是比较接近的, 而对边界点或者非圆盘中心角点, 则其中心对称的像素点的色相值 h 却有较大的差异。图2为彩色圆盘的4个不同特殊方位。

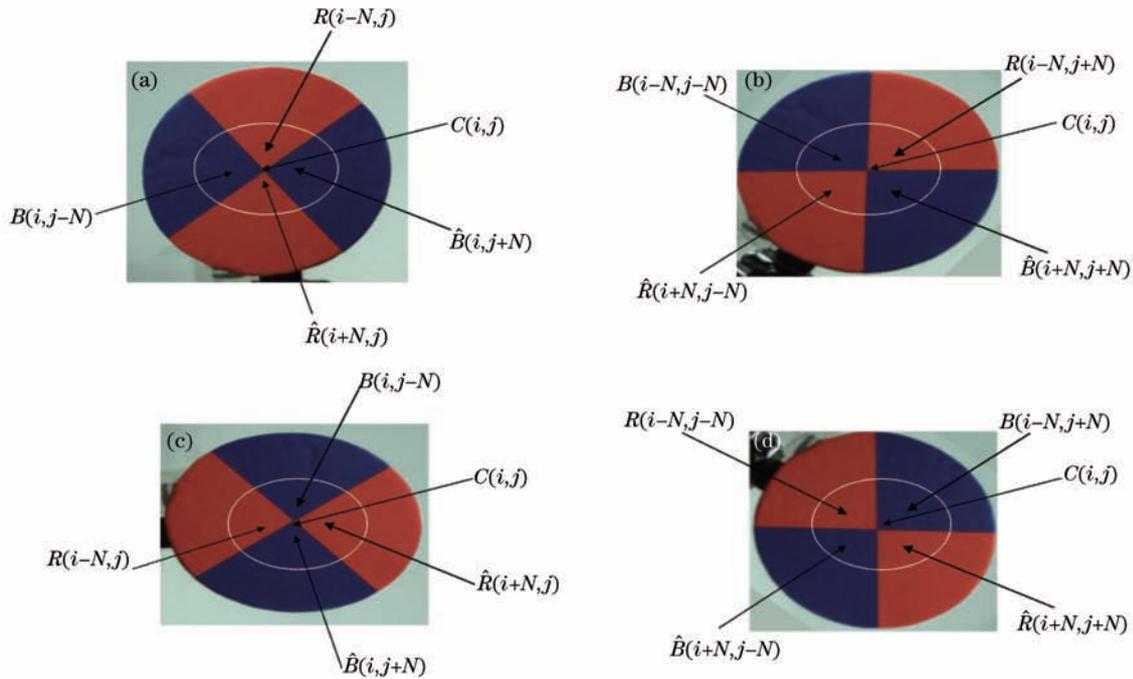


图2 (a)圆盘上下红色,左右蓝色;(b)左下、右上红色,左右蓝色;(c)上下蓝色,左右红色;(d)左上、右下蓝色,左上、右下红色
Fig.2 (a) Upper and down sides of the disc are red, left and right are blue; (b) lower left and upper right are red, upper left and lower right are blue; (c) upper and down are blue, left and right are red; (d) lower left and upper right are blue, upper left and lower right are red

在图2中, R 与 \hat{R} 是关于中心点 C 中心对称的红色部分, B 与 \hat{B} 是关于中心点 C 中心对称的蓝色部分。对于彩色圆盘中心角点的检测, 需要考虑到圆盘放置的所有位置, 提高图像角点检测的自适应能力, 图2(a)~(d)是平面上 $0^\circ \sim 180^\circ$ 间4个特殊的位置, 根据本文提出的算法, 利用4种不同的对称算子检测圆盘的中心角点, 针对上述4种极限条件都能够检测出来, 那么在它们之间旋转的任意角度也都能够检测出来。同时彩色圆盘的对称性, 圆盘在 $180^\circ \sim 360^\circ$ 间也就有相同变化。所以彩色圆盘在 $0^\circ \sim 360^\circ$ 旋转的任意角度都可对中心角点进行检测。

2.2 算法设计和实现步骤

根据2.1中对红蓝彩色对称圆盘的分析, 具体的算法设计和实现步骤如图3所示。

基于对称算子的红蓝彩色圆盘角点检测方法, 根据其特征, 具体步骤如下。

1) 在现实中, 用摄像机获取的图像往往都存在着噪声, 所以在图像处理前需要对图像进行滤波处理。在进行数学仿真或误差评估时, 往往认为传感器所引入的噪声服从正态分布(高斯白噪声), 所以选用高斯函数来对图像进行滤波。图像高斯滤波的思路是: 对高斯函数进行离散化, 以离散点上的高斯函数值为权重, 对采集到的图像矩阵的每个像素点做一定范围内的加权平均, 即可有效消除高斯噪声。离散的高斯卷积核 G 选用 $(2k+1) \times (2k+1)$ 维。高斯滤波的具体表达式为

$$\begin{cases} I(i,j) = I_0(i,j) * G(u,v) = \sum_{u=-k}^k \sum_{v=-k}^k I_0(i+u, j+v) \cdot G(u,v) \\ G(u,v) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{u^2+v^2}{2\sigma^2}\right) \end{cases}, \quad (4)$$

式中 (i,j) 为像素的坐标值, $I_0(i,j)$ 为原始输入图像在 (i,j) 处的灰度值, $I(i,j)$ 为处理后的图像在 (i,j) 处的灰度值, $G(u,v)$ 为高斯核函数在 (u,v) 处的归一化值, σ 为高斯函数尺度值, 离散化的高斯函数可由连续高斯函数

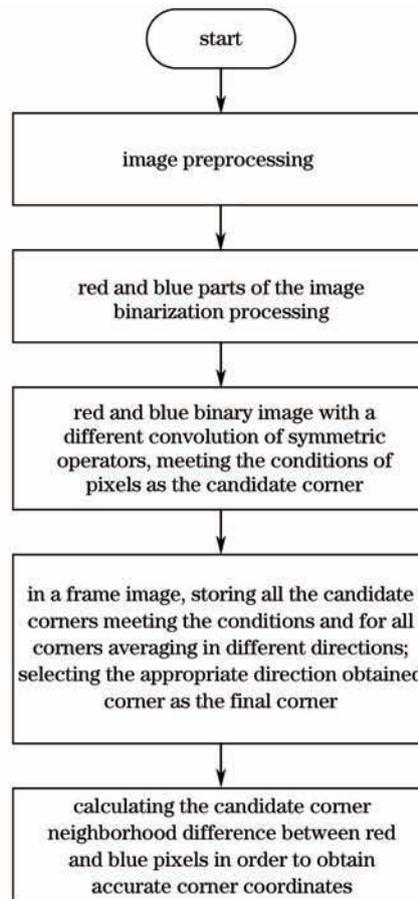


图3 总体设计流程图

Fig.3 Overall design flow diagram

采样得到。

2) 根据红蓝彩色色相值 h 的不同, 提取出红色二值化图像 $I_R(i,j)$ 和蓝色二值化图像 $I_B(i,j)$, 其中红色部分 $h \in [350^\circ, 360^\circ) \cup [0^\circ, 10^\circ]$, 蓝色部分 $h \in [230^\circ, 250^\circ]$ 。4种不同方位下的红蓝二值化图像如图4所示。

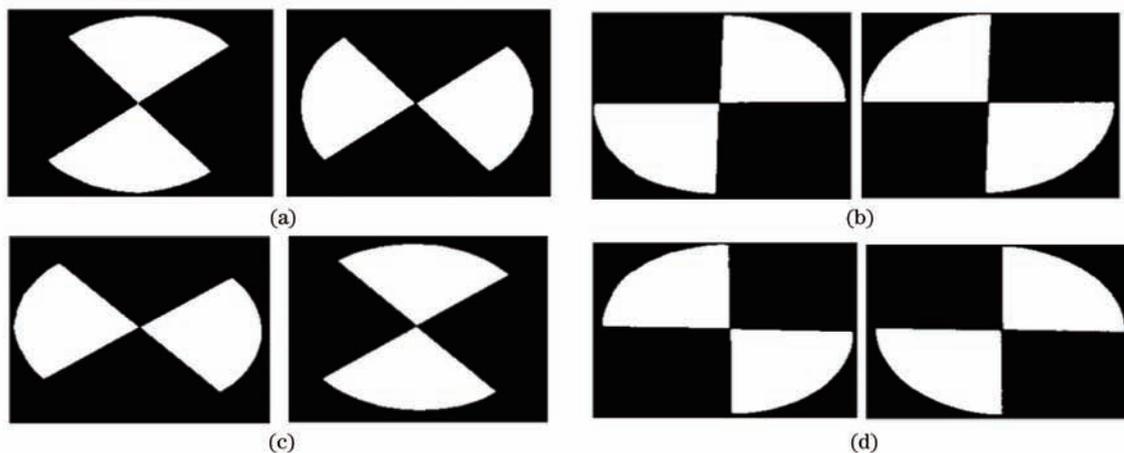


图4 对应图2方位的红色(左)和蓝色(b)二值化图像

Fig.4 Binarized images for red (left) and blue (right) parts of the cases shown in Fig.2

从上面4组二值化分割后的图像可知, 用HSV对颜色进行分割后特征非常明显, 对周围环境的其他彩色信息具有良好的抑制效果。

3) 根据得到的不同红蓝彩色二值化图像, 分别采用 $(2N+1) \times (2N+1)$ 纵向对称算子 H 、横向对称算子 H^T 、倾斜 45° 下的纵向对称算子 S 和倾斜 45° 下的横向对称算子 S^T 对应4种情形。其中对称算子 H 、 H^T 、

S 、 S^T 如下所示,“ \dots ”表示多行多列0, N 可根据实际情况而定,本文检测的算法中 N 取10。

$$H = \begin{bmatrix} \dots & \dots & 1 & 1 & 1 & \dots & \dots \\ \dots & \dots & \dots & 1 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & 0 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & 1 & \dots & \dots \\ \dots & \dots & 1 & 1 & 1 & \dots & \dots \end{bmatrix}, \quad (5)$$

$$H^T = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & \dots & \dots & \dots & \dots & \dots & 1 \\ 1 & 1 & \dots & 0 & \dots & 1 & 1 \\ 1 & \dots & \dots & \dots & \dots & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix}, \quad (6)$$

$$S = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & 0 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & \dots & \dots & \dots & \dots & \dots \end{bmatrix}, \quad (7)$$

$$S^T = \begin{bmatrix} 1 & 1 & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & 0 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & 1 & 1 \end{bmatrix}, \quad (8)$$

彩色圆盘的角点处于红蓝中心的交界处,对于对称的红蓝扇形图案,除了要满足在红蓝的交界处外,还必须满足对称算子在相同的区域处要有相似的颜色,之所以要用倾斜 45° 的对称算子 S 、 S^T ,是为了防止圆盘处在图2(b)、(d)下的特殊条件下 H 、 H^T 算子不能够很好地检测中心角点。

具体算法如下。用(5)~(8)式中的 H 、 H^T 、 S 、 S^T 对称算子分别对红色二值化图像 $I_R(i,j)$ 卷积得到响应值函数 $R_{R_H}(i,j)$ 、 $R_{R_{H^T}}(i,j)$ 、 $R_{R_S}(i,j)$ 、 $R_{R_{S^T}}(i,j)$ 和蓝色二值化图像 $I_B(i,j)$ 卷积得到响应值函数 $R_{B_H}(i,j)$ 、 $R_{B_{H^T}}(i,j)$ 、 $R_{B_S}(i,j)$ 、 $R_{B_{S^T}}(i,j)$ 。具体表达式如下:

$$R_{R_H}(i,j) = I_R(i,j) * H = \sum_{m=-N}^N \sum_{n=-N}^N I_R(i+m,j+n) \cdot H(m+N+1, n+N+1), \quad (9)$$

$$R_{R_{H^T}}(i,j) = I_R(i,j) * H^T = \sum_{m=-N}^N \sum_{n=-N}^N I_R(i+m,j+n) \cdot H^T(m+N+1, n+N+1), \quad (10)$$

$$R_{R_S}(i,j) = I_R(i,j) * S = \sum_{m=-N}^N \sum_{n=-N}^N I_R(i+m,j+n) \cdot S(m+N+1, n+N+1), \quad (11)$$

$$R_{R_{S^T}}(i,j) = I_R(i,j) * S^T = \sum_{m=-N}^N \sum_{n=-N}^N I_R(i+m,j+n) \cdot S^T(m+N+1, n+N+1), \quad (12)$$

$$R_{B_H}(i,j) = I_B(i,j) * H = \sum_{m=-N}^N \sum_{n=-N}^N I_B(i+m,j+n) \cdot H(m+N+1, n+N+1), \quad (13)$$

$$R_{B_{H^T}}(i,j) = I_B(i,j) * H^T = \sum_{m=-N}^N \sum_{n=-N}^N I_B(i+m,j+n) \cdot H^T(m+N+1, n+N+1), \quad (14)$$

$$R_{B_S}(i,j) = I_B(i,j) * S = \sum_{m=-N}^N \sum_{n=-N}^N I_B(i+m,j+n) \cdot S(m+N+1, n+N+1), \quad (15)$$

$$R_{B_{S^T}}(i,j) = I_B(i,j) * S^T = \sum_{m=-N}^N \sum_{n=-N}^N I_B(i+m,j+n) \cdot S^T(m+N+1, n+N+1), \quad (16)$$

分别对红蓝二值化求取响应最大值,得到有效的候选角点。为了全方位地对圆盘进行实时角点检测,需要对上述圆盘放置的4种特殊位置逐一进行讨论,将满足条件的像素点的坐标值 (i,j) 储存下来,最后对所

有满足条件的像素点求平均,得到最终的候选角点坐标 (i_0, j_0) 。

对于圆盘的放置为上下红色,左右蓝色,如果 $R_{R_n}(i,j)=8$, $R_{B_n}(i,j)=8$,则

$$\begin{cases} i(N_1) = i \\ j(N_1) = j \\ N_1 = N_1 + 1 \end{cases}, \quad (17)$$

当一帧图像结束后,对所有满足条件的像素点求平均,并把计数值 N_1 清零。得到最终候选角点的坐标值为

$$\begin{cases} i_{01} = \frac{\sum_{n=0}^{N_1-1} i(n)}{N_1} \\ j_{01} = \frac{\sum_{n=0}^{N_1-1} j(n)}{N_1} \end{cases}. \quad (18)$$

对于圆盘的放置为圆盘倾斜 45° (左下、右上红色,左上、右下蓝色),如果 $R_{R_s}(i,j)=8$, $R_{B_s}(i,j)=8$,则

$$\begin{cases} i(N_2) = i \\ j(N_2) = j \\ N_2 = N_2 + 1 \end{cases}. \quad (19)$$

当一帧图像结束后,对所有满足条件的像素点求平均,并把计数值 N_2 清零。得到最终候选角点的坐标值为

$$\begin{cases} i_{02} = \frac{\sum_{n=0}^{N_2-1} i(n)}{N_2} \\ j_{02} = \frac{\sum_{n=0}^{N_2-1} j(n)}{N_2} \end{cases}. \quad (20)$$

对于圆盘的放置为上下蓝色,左右红色,如果 $R_{R_n}(i,j)=8$, $R_{B_n}(i,j)=8$,则

$$\begin{cases} i(N_3) = i \\ j(N_3) = j \\ N_3 = N_3 + 1 \end{cases}. \quad (21)$$

当一帧图像结束后,对所有满足条件的像素点求平均,并把计数值 N_3 清零。得到最终候选角点的坐标值为

$$\begin{cases} i_{03} = \frac{\sum_{n=0}^{N_3-1} i(n)}{N_3} \\ j_{03} = \frac{\sum_{n=0}^{N_3-1} j(n)}{N_3} \end{cases}. \quad (22)$$

对于圆盘的放置为圆盘倾斜 45° (左下、右上蓝色,左上、右下红色),如果 $R_{R_s}(i,j)=8$, $R_{B_s}(i,j)=8$,则

$$\begin{cases} i(N_4) = i \\ j(N_4) = j \\ N_4 = N_4 + 1 \end{cases}. \quad (23)$$

当一帧图像结束后,对所有满足条件的像素点求平均,并把计数值 N_4 清零。得到最终候选角点的坐标值为

$$\begin{cases} i_{04} = \frac{\sum_{n=0}^{N_4-1} i(n)}{N_4} \\ j_{04} = \frac{\sum_{n=0}^{N_4-1} j(n)}{N_4} \end{cases}. \quad (24)$$

以上各式中的 N_1 、 N_2 、 N_3 、 N_4 分别为相应条件下一帧图像满足条件的候选角点数量, $i(N_1)$ 、 $i(N_2)$ 、

$i(N_3)$ 、 $i(N_4)$ 分别对应像素点行坐标 i 的存储, $j(N_1)$ 、 $j(N_2)$ 、 $j(N_3)$ 、 $j(N_4)$ 分别对应像素点列坐标 j 的存储, (i_{01}, j_{01}) 、 (i_{02}, j_{02}) 、 (i_{03}, j_{03}) 、 (i_{04}, j_{04}) 分别为相应条件下的最终候选角点的行、列坐标值。

由上面的分析可知,对上面 4 种情况的准确检测,即可把 $0^\circ \sim 360^\circ$ 内圆盘放置的任意位置处的中心角点检测出来,但针对圆盘放置的某一位置,可能不止满足其中一种情况,为了更好地提取中心角点,将 4 种情况中满足条件最多的像素点 N 作为最终的结果,并把其相应的候选角点的坐标值作为最终的中心角点坐标值 (i_0, j_0) 。

4) 对步骤 3) 得到的候选点进行正确性检验。检验的过程如下:建立一个一定大小的窗口 W (窗口大小如 5 pixel×5 pixel) 如图 14 所示,以角点 $I(i_0, j_0)$ 为中心,比较窗口内红色像素点的个数 N_R 与蓝色像素点个数 N_B 的差值,如果 $|N_R - N_B| < T_{th}$ (T_{th} 为某一阈值,对于 5 pixel×5 pixel 的窗口, T_{th} 可取 13),则该点为正确的角点;如果不满足,则该点为错误的角点。

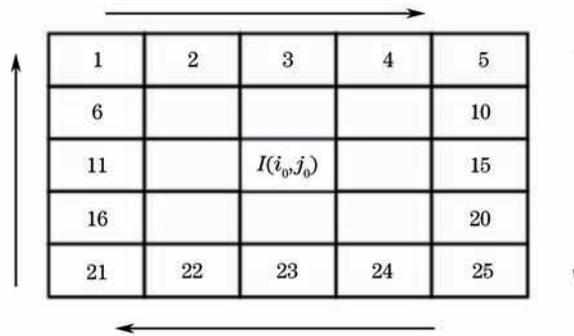


图 5 5 pixel×5 pixel 窗口区域

Fig.5 5 pixel×5 pixel square window

在上面的步骤中,对角点可以先求取最终的平均值,再进行准确性检验,也可以对满足条件的像素点先进行正确校验,再对正确的进行求平均,作为最后的角点,但计算量会增大。

3 实验结果及分析

为了验证该方法的有效性和准确性,采用 Basler aca2040-180kc 的 Camera link 相机拍摄出 50 frame/s, 大小为 2048 pixel×2048 pixel 的红蓝彩色圆盘图像进行检测,通过 Matlab 仿真,与经典的 Harris 算法^[1]进行了对比,在软件上验证红蓝彩色圆盘定标的准确性和可靠性,再将程序移植到基于 ALTERA 公司的 Cyclone IV EP4C115F29C7N 的 FPGA 的平台上进行硬件实时性仿真,证明了该算法在硬件中易实现。

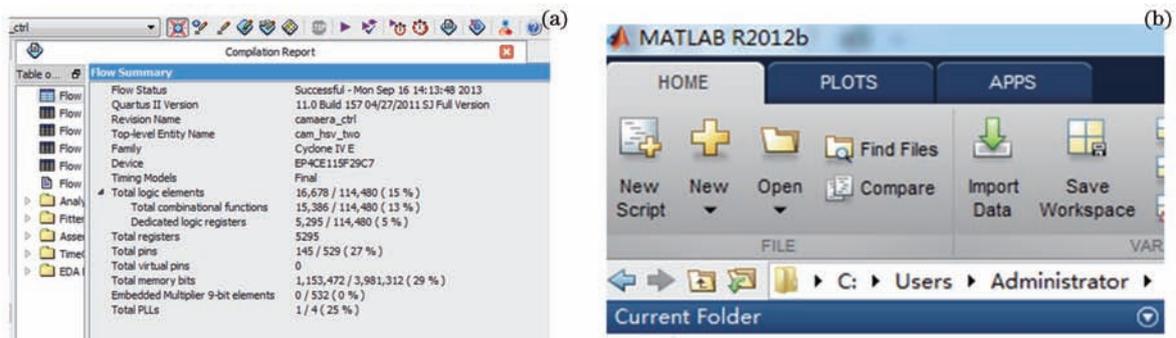


图 6 软硬件控制界面。(a) FPGA 硬件控制界面; (b) Matlab 软件控制界面

Fig.6 Software and hardware control interface. (a) FPGA hardware control interface; (b) Matlab software control interface

首先进行 Matlab 软件仿真实验。由于实际成像时边缘和角点出可能发生模糊,为此在软件仿真时,为了算法能在不同的实际环境中都具有良好的稳健性,首先对红蓝彩色对称圆盘加入均值为 0、方差为 0.01 的高斯噪声来模拟一幅干扰的图片,根据图 2 的 4 个不同方位,得到两种方法下检测到的角点图,如图 7 所示。

图 7 都是经过局部放大后的检测图像,其中绿色的点表示圆盘中心点,可以看到 Harris 算法检测到的角

点产生了偏移。对于不清晰图像, Harris算法检测到的角点值会出现在角点周围模糊处。表1分别记录了上述圆盘4个不同方位下中心角点的数据, 并与真实值和 Harris 检测值对比。

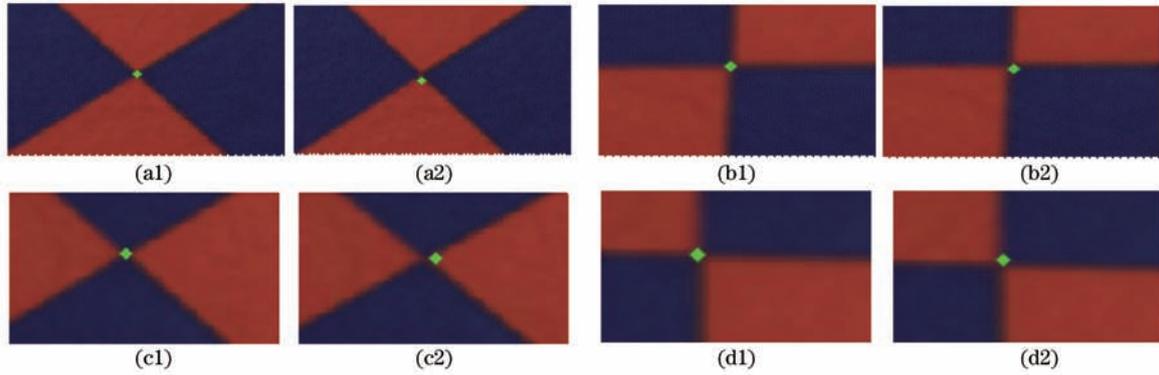


图7 对应图2方位的采用本文方法(a1),(b1),(c1),(d1)和Harris算法(a2),(b2),(c2),(d2)检测的结果

Fig.7 Detection results obtained by the proposed method (a1), (b1), (c1), (d1) and Harris algorithm (a2), (b2), (c2), (d2) for the cases shown in Fig.2

表1 仿真图角点检测数据对比(单位: pixel)

Table 1 Simulation image corner detection data comparison (unit: pixel)

True value	(112,132)	(103,125)	(100,129)	(101,128)
Harris algorithm	(114,132)	(105,126)	(102,129)	(103,129)
Proposed algorithm	(112,132)	(104,125)	(100,129)	(102,128)

表2给出了使用Harris算法和本文方法完成角点检测的时间, 可以看到, 本文的算法比Harris算法快得多。

表2 算法消耗的时间

Table 2 Time consumption comparison

Algorithm	Elapsed time /ms
Harris algorithm	6160
Proposed algorithm	465

Harris算法作为经典的通用算法, 检测图像区域内曲率最大值处作为图像的角点。该算法通过移动一个小图像窗口来观察区域内像素灰度变化从而识别角点, 为便于计算, 一般移动窗口选用高斯窗口函数, 计

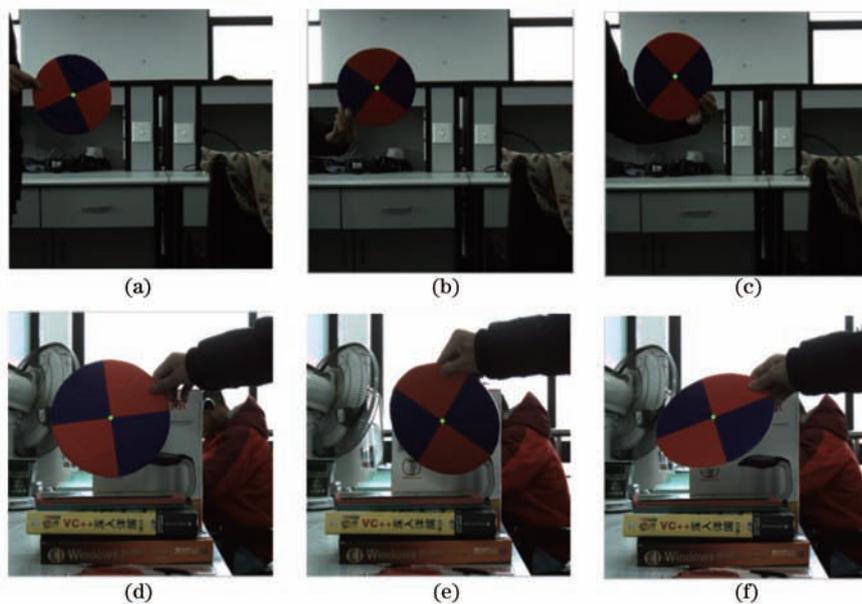


图8 不同场景, 伸缩、旋转和尺度下的角点检测结果

Fig.8 Stretching, rotation and scale corner detection results under different scenarios

算时需要用到高斯滤波函数、图像行列方向上的梯度,且需计算相应特征矩阵的行列式和矩阵的迹,因此其计算量较大,计算时间较长,且在有些条件下误差较大。图8是不同场景,伸缩、旋转和尺度下对彩色圆盘角点检测的结果,绿色的点为检测到的中心角点。

图9是基于Altera FPGA的硬件设计框图,图10为硬件验证平台,该平台很好地验证了该算法在硬件上的可行性和实时性。

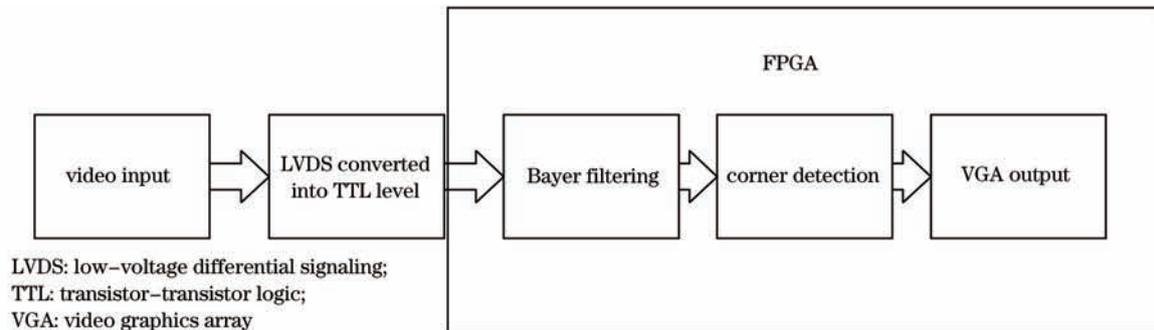


图9 FPGA设计框图

Fig.9 FPGA Design diagram

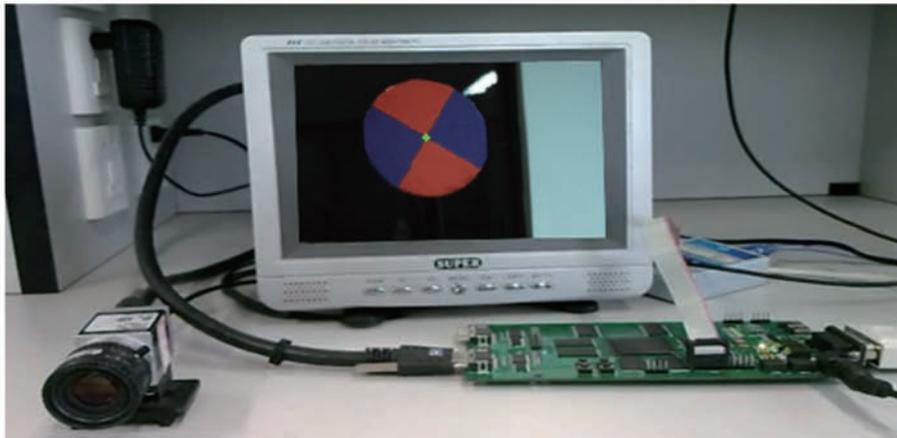


图10 基于FPGA的硬件平台

Fig.10 FPGA hardware platform

4 结 论

提出了一种新的圆盘角点检测方法,并将其与经典的Harris角点检测法^[1]进行了比较。仿真及实验结果表明:该算法构思巧妙、计算量小,易于硬件化实现,不仅显著提高了图像角点检测的速度,并且有效地避免了错误角点的产生。通过对图像不同场景下的仿真测试,证明了该方法对图像旋转、伸缩和尺度变换下的角点检测都有很好稳健性,因而能够很好地应用于无人机侦查任务中目标的定位。

参 考 文 献

- 1 Harris C, Satephens M J. A combined corner and edge detector [C]. 4th Alvey Vision Conference, 1988. 147-152.
- 2 He Jianwei, Yang Jianfeng, Xue Bin, *et al.*. Research on the automatic stitching of panorama camera image based on system similarity matching principle [J]. Acta Optica Sinica, 2010, 30(4): 989-993.
何建伟, 杨建峰, 薛 彬, 等. 基于系统论匹配准则的尺度不变特征变换的图像自动拼接研究[J]. 光学学报, 2010, 30(4): 989-993.
- 3 Zhang Tienan, Tu Yaqing Wang Dechao. Novel algorithm for fast detecting checkerboard corners [J]. Computer Engineering and Applications, 2008, 44(31): 236-238.
张铁楠, 涂亚庆, 王德超. 针对棋盘格角点快速检测的一种新方法[J]. 计算机工程与应用, 2008, 44(31): 236-238.
- 4 Tian Feng, Yan Jianguo, Cao Yinghui. Research on target location of UAV video [J]. Computer Measurement and

- Control, 2010, 18(4): 900–901.
田 丰, 闫建国, 曹莹慧. 无人机视频中的目标定位研究[J]. 计算机测量与控制, 2010, 18(4): 900–901.
- 5 Wang Xiangjun, Wang Yan, Li Zhi. Fast target recognition and tracking method based on characteristic corner [J]. Acta Optica Sinica, 2007, 27(2): 360–365.
王向军, 王 研, 李 智. 基于特征角点的目标跟踪和快速识别算法研究[J]. 光学学报, 2007, 27(2): 360–365.
- 6 Yu Mei, Yu Bokang. Indicatory sign detection based on color image [J]. Computer Engineering and Applications, 2000, 36(4): 166–168.
郁 梅, 郁伯康. 基于彩色图像的指示标志检测 [J]. 计算机工程与应用, 2000, 36(4): 166–168.
- 7 Wang Tao, Sun Changku, Shi Yongqiang, *et al.*. Novel grating projection system based on assistant line and its calibration method [J]. Acta Optica Sinica, 2011, 31(1): 0115002.
王 涛, 孙长库, 石永强, 等. 基于辅助参考线的光栅投影轮廓测量系统及标定方法[J]. 光学学报, 2011, 31(1): 0115002.
- 8 Tian Yuanyuan, Huang Hecheng, Tan Qingchang, *et al.*. Size measurement of parts based on machine vision [J]. Laser & Optoelectronics Progress, 2010, 47(1): 011501.
田原嫫, 黄合成, 谭庆昌, 等. 基于机器视觉的零件尺寸测量[J]. 激光与光电子学进展, 2010, 47(1): 011501.
- 9 Hou Junjie, Wei Xinguo, Sun Junhua. Calibration method for binocular vision based on matching synthetic images of concentric circles [J]. Acta Optica Sinica, 2012, 32(3): 0315003.
侯俊捷, 魏新国, 孙军华. 基于同心圆合成图像匹配的双目视觉标定[J]. 光学学报, 2012, 32(3): 0315003.
- 10 Shi Meihong, Shen Liang, Long Shizhong. From RGB to HSV color space conversion formula of correction [J]. Basic Sciences J Textile Universities, 2008, 21(3): 351–355.
石美红, 申 亮, 龙世忠. 从RGB到HSV色彩空间转换公式的修正[J]. 纺织高校基础科学学报, 2008, 21(3): 351–355.