A fast and sub-pixel detector for grid-like target in camera calibration

Le-jun Shen[1,2]
Department of Sport Information,
Chengdu Sport University
Chengdu, PR China
sljcool@hotmail.com

Zhun-yu Ke[1]
Institute of Image & Graphics, College of Computer,
Sichuan University
Chengdu, PR China

Abstract: **Sub-pixel detection of target points is the performance bottleneck in camera calibration. Traditional algorithms are computational expensive or low precision when we do camera calibration in sport video analysis. In this paper, we propose a new algorithm to detect the grid-like target (i.e. tennis court in TV broadcasting). It has 3 parts: (1) color histogram based interested point classifier making our method faster; (2) sub-pixel refinement by non-linear least squares method improving the accuracy; (3) extended line scan using interested point as the start/end point finding the final line parameters. Results indicate that our detector is faster (<9ms), more accurate and requires less memory than Hough based algorithms if target is grid-like: "straight lines link together".**

Keywords: Hough transform; sub-pixel; model-based video analysis; Sport video processing; camera calibration;

I. INTRODUCTION

The tennis court is a grid-like target for camera calibration in sports video analysis: semantic analysis of sport video [1], virtual advertisement insertion [2], tactics analysis [3] and so on. The state-of-the-art Hough transform was used in grid-like target detection. But it is less effective in computing since it uses all the sample points or all the point pairs to vote. Many researchers improved performance of Hough transform. Yu[4] uses the fact "the straight lines in sports video are sparse and long" to extract soccer court lines fast. Farin [5] apply a real-time RANSAC-based line detector to obtain line segments of tennis court line.

Sub-pixel accurate is another topic in this paper. Chen [6] found a sub-pixel detection algorithm of artificial grid target. We mainly concern on natural grid-like target (i.e. tennis court in broadcast video stream). **The main challenge** is that camera changes dramatically, not smoothly, in broadcasting. The virtual line intersection points are important because they are calibration points [5][7]. The sub-pixel positions are computed from the line parameters.

This paper presents a fast (<9ms) and sub-pixel detection algorithm for grid-like target in camera calibration. Grid-like means target has the property: "straight lines link together". **The contributions** of this paper are:

(1) We provide a hybrid solution to "multi-pixel in a line" problem (Figure 1). Many other papers refine a bunch of line candidates by robust technology, such as RANSAC or LTS [5]. We develop a new framework to get sub-pixel accurate line parameters at very low computational cost.

(2) We propose a sub-pixel refinement method for grid-like target, which computes the minimal of a gradient model using non-linear least squares method. It is proved to be more accurate and efficient than traditional ones.

(3) We remove false alarms by two technologies: "hue + white-pixel" histogram and "edge-to-area" ratio. It is proved to be more robust, more effective and more distinctive than traditional color histogram.

II. ALGORITHM DETAILS

A. Color aided interested point classifier

The *interested points* are image feature points near the grid

(a) (b) (c) (d) (e) (f)

Figure 1: "Multi-pixel in a line" problem. (a) It is a very simple grid target. Each line is 3 pixels width. (b)One grid line has two parallel Canny edges. (c) Standard Hough transform (SHT) gets 2 line candidates (red color) every line. (d) In tennis video, however, SHT gets 4~7 candidates every line because of "Multi-pixel in a line" problem: Hough transform find too many local peaks in parameter space. (e)Progressive Probabilistic Hough transform (PPHT) [8] gets 2~4 candidates. (f) We provide a fast and sub-pixel accurate solution to this problem and grid-like target (i.e. tennis court) can be detected correctly.

line (i.e. court line). They have some distinctive characteristic from others: (almost) same background color, near the white-color line and sparse and long court lines. In this section, we apply some quick test to reject non-interested point.

After the Harris corner detection [9], we get about 250~450 image point shown in Fig. 2(a). If all of them are directly used in the sub-pixel refinement without filtering, computing time is up to 40ms. In many computer vision applications, color-based techniques play an important role because they provide useful information at very low computational cost. Hence, we use the HSV color space and color histogram[10] to describe the distinctive characters and divide point set into two classes: interested or none-interested.

For fast computing, only Hue is used to construct histogram with 180 bins. To compensate the lost information (i.e. Saturation), we add small amount extra bins into histogram.

Definition 1: “hue + white-pixel” histogram. Let H is 1-D histogram, H_{index} = Hue*(1-W) + 182*W. W is white-pixel image[5]. Pixel value in W is "1" if it is white color and "0" for others.

The color probability distribution can be visualized. Fig. 2(b) shows 7 instances. To find the most possible court lines, the classifier is designed for 0% false negative rate. So, false positive rate is very high and classifiers rarely miss detection. As shown in Fig.3, "hue+ white-pixel" histogram improves classifier performance significantly.

Definition 2: "edge-to-area" ratio. Let A is area of rectangle, E is count of edge pixels in rectangle. "Edge-to-area" ratio R_{ea} =E/A. Note that A=17*17=289 in our experiment.

Court lines are always sparse and long in natural images. So, edge-to-area ratio excludes some points in textured area or highly structured area (Fig. 4). This technology also cut down false positive rate (purple curve in Fig. 3).

Definition 3: interested point. A Harris corner p_i is labeled as interested point if $D_{BHATTACHARYYA}$ $(h_i, g_j) < t_j$ and $R_{ea}(p_i) < T$, where h_i is “hue + white-pixel” histogram ($i<N$), g_j is mean of sub-class, t_j is the associated threshold ($j<M$) and R_{ea} is "edge-to-area" ratio.

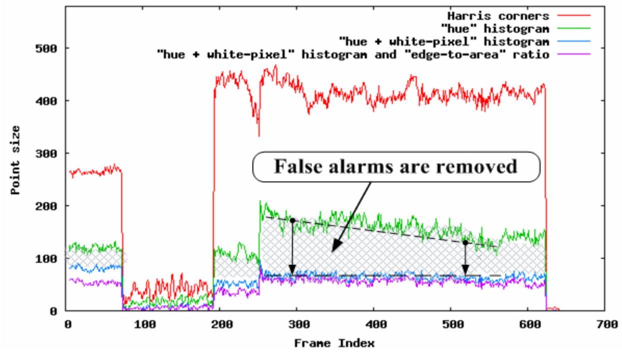


Figure 3: Compared with hue histogram, "Hue + white-pixel" histogram remove nearly 37%~65% false alarms (area between green and blue curve) at very low computational cost.

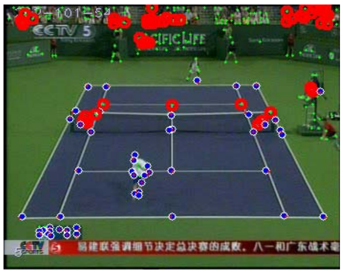
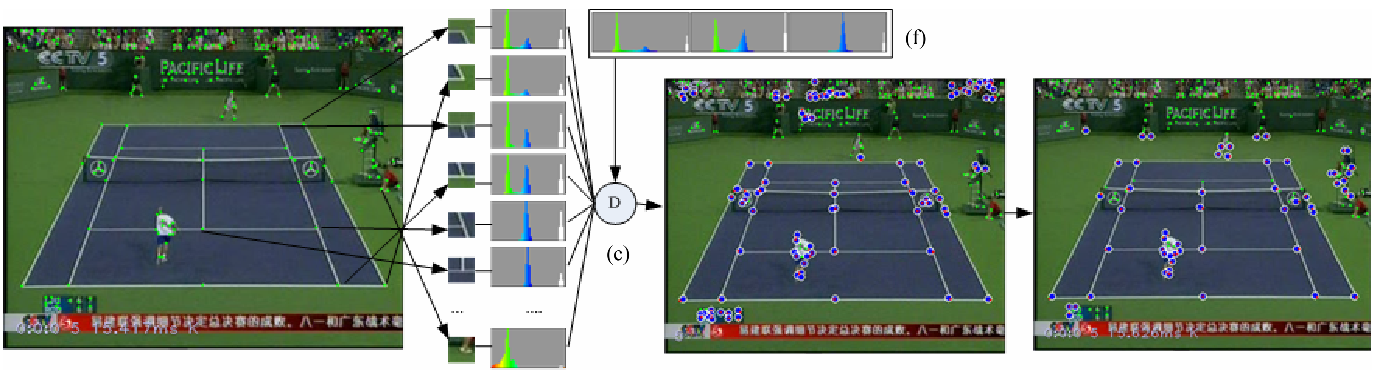


Figure 4: Red circles in image are labeled as high textured points if "edge-to-area " ratio are larger than threshold. They are removed as false alarms.

This classifier is simple and efficient. Firstly, about 85% points are classified as non-interested points, which will not be processed in the next steps. So, this classifier significantly reduces computing cost of non-linear optimization in the next step. Secondly, the “hue + white-pixel” histogram is invariant to camera rotating, zooming and panning. It is also invariable to light condition in long time sport video. Thirdly, the performance of the classifier is surprisingly good in our experiment. The robustness is achieved by two technologies: "hue + white-pixel" histogram and "edge-to-area" ratio. Note that 37 false alarms are removed from Fig. 2(d) to Fig. 2(e).



(a) (b) (d) (e)

Figure 2: Details of color based interested point classifier. (a) 264 Harris corners (green dots in image). (b) Extract 264 image patches center on Harris corners and compute each “hue + white-pixel” histogram. Note that the most right white color bar is white-pixel bin. (c) Compute BHATTACHARYYA distance between observation and means of sub-class. (d)101 points (blue-white circles in image) are classified as interested point if their distance below threshold. (e) 64 points are finally accepted using both "hue + white-pixel" histogram and "edge-to-area" ratio. (f) 3 means of sub-class from learning stage.

B. Sub-pixel refinement using Levenberg -Marquardt method

The interested point position of Harris corner cannot be directly used as the start and end point in the next step (i.e. line scan). There is 1~3 pixel distances from Harris corner to the center of intersection point of lines. That means the 0.03~0.18m error in world coordination after camera calibration. Inspired by OpenCV, we use the image gradient model to get sub-pixel position.

Definition 4: sub-pixel position. Given a shift (Δu, Δv) and a point (u, v) in the image, $g=(gx, gy)^T$ is the image gradient at point (u+Δu, v+Δv). $x=(u, v)^T$ is parameter. $v=(\Delta u, \Delta v)^T$ is shift vector. The cost function is defined as:

$$F(x) = \sum_{v \in W} (g^T v (1 - e^{-(\Delta u^2 + \Delta v^2)/a}))^2 \qquad (1)$$

Where W denotes 11*11 window centered on x. The sub-pixel position of interested point is the minimal of F(x) (Fig. 5).

$$X^* = \arg\min_x \{F(x)\} \qquad (2)$$

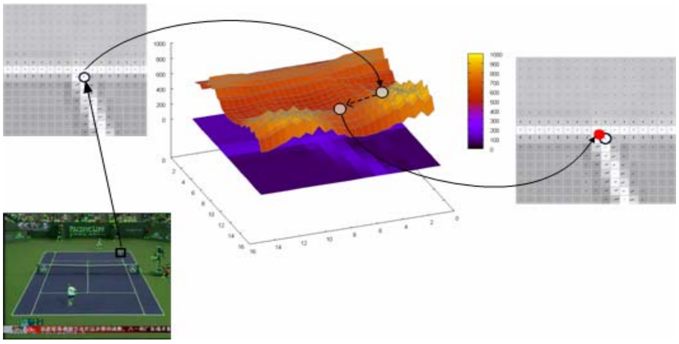


Figure 5: Sub-pixel position. **Bottom left**: Source image. **Top left**: Rectangle near the Harris corner (243, 88). **Center**: Surface of F(x) near this point. Dashed line is the optimization process from start position (243, 88) to the minimal (242.32, 87.51). **Top right**: The minimal is drawn as a red dot. It is a better location as intersection point of lines than Harris corner.

If v is orthogonal to the image gradient g, $g^T v$ equal to 0. The more points within local neighborhood of x orthogonal to the gradient, the smaller value of the summation of $g^T v$ is. The minimal of F(x) is the center of intersection point of straight lines. It can be found by non-linear least squares method. Levenberg-Marquardt method is implemented in our system.

The reason why multiply weighting factor $(1 - e^{-(\Delta u^2 + \Delta v^2)/a})$ in equation is: the image gradient is ambiguous when both eigenvalues of Hessian matrix are high (Example is Harris corner (243, 88) in Fig. 5). These ambiguous pixels should have a very low weight. To simplify computing, we assign the region in the middle of window W a very low weight. It is proved that this weighting factor makes the F(x) more stable and X* more accurate. The factor "a" is chosen by user and associated with expected width of court lines in images. This weighting factor can be pre-computed before optimization in order to get higher performance.

C. Line scans using interested point as the start/end point

There are intersection points of grid lines in grid-like target. Intersection point set is part of interested point set. If we select the start-end point pairs iteratively from interested point set, we can find court lines by line scan. One by one searching among point set means the time complexity is $O(n^2)$. In practice, some candidate lines are fast rejected when their length is shorter than threshold. We do not interest in short lines in sports video because they cannot provide accurate information for camera calibration[5].

There is "data missing" problem using interested point as the start-end point pair. Every pixel in image has been voted in Hough transform. So, court line L near the image border can be detected correctly. Global information collected by Hough transform guarantee all possible straight lines can be found. However, line scan are relying on the start-end points pair, which collect only local information. In Fig. 6, L misses both the start and end point. Without the start-end point pair, we can not detect line L correctly. In other word, local information is not sufficient if part of court is not visible in image when camera zooming or panning.

Extended line scan can solve this problem, demonstrated in Fig. 6 and Algorithm 1. That is to say, we can find the whole line if more than 2 interested points exist on the line.

The line L is defined by two points (S and E in Fig. 6). After scanning along the line L in white pixel image, we obtain a pixel sequence $S(x)=\{s_1,s_2,\ldots,s_n\}$. Value of S(x) is the pixel value (0 or 1) at position x in W. Operator "**x++**" means the next point along the direction from S to E on the line L.

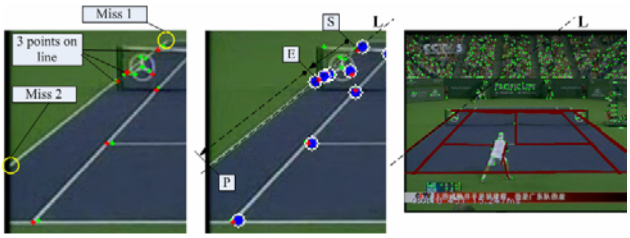


Figure 6: "Data missing" problem are fixed by extended line scan. **Left**: Court line start point and end point are missing because of camera zooming. "Miss 1" is caused by very low Harris corner response. "Miss 2" is caused by the border effect (image gradient near the image border makes the sub-pixel refinement failure). Only 3 points in the middle of line exist. **Middle**: There are 3 points exist in the middle of L. After line scanning from start point "S" to end point "E", we extend (prolong) scan to "P" until the white-pixel is unavailable. **Right**: Line scanning result by Algorithm 1. Note L is detected correctly even the start/end point is missing.

Algorithm 1 Extend line scan

Input: RGB image I, white pixels image W and the set of points $P=\{p_1,p_2,\ldots p_K\}$;
Output: the set of lines $\Gamma=\{l_1,l_2,\ldots,l_Z\}$.

$\Gamma=\Phi$; Z=0; $\tau_1=5$; $\tau_2=4$; $\tau_3=3$; $\tau_4=25$;
For(i=1;i<K;i++){
  For(j=i+1;j<K;i++){
    if ($D_{EUCLIDEAN}(p_i, p_j) > \tau_1$)
      continue;
    Get the sequence S(x) along the line defined by two points (p_i, p_j) from image W.
    S= p_i; E= p_j; x=S; e=S; gap=0;

```
        while (x<>E or gap>τ2 ){
           if (S(x)=1){
              e=x; gap=0; x++;
           }else{
              gap=gap+1;
           }
        }
        if (x=E){
           while (S(x)<> s1 or S(x)<> sn or gap>τ3){
              if (S(x)=1){
                 e=x; gap=0; x++;
              }else{
                 gap=gap+1;
              }
           }
        }
        if (D_EUCLIDEAN(S,e) >τ4){
           Γ←Γ+{ (S,e)}; Z=Z+1;
        }
  }}
```

III. EXPERIMENT AND RESULT

The testing videos (42 minutes, 320*240) are captured from CCTV-5. The video is ATP Masters Cup Championships 2008, which is the game between Nadal and Roddick. All algorithms are implemented using C++ and compiled in Visual Studio 2005. The experiments were performed on P4 2.9GHz with 512M RAM.

In broadcast video, the camera changes dramatically, including panning, zooming and multi-camera switching. In most case, our algorithm detects tennis court correctly. The results at different camera view are illustrated in Fig. 7.

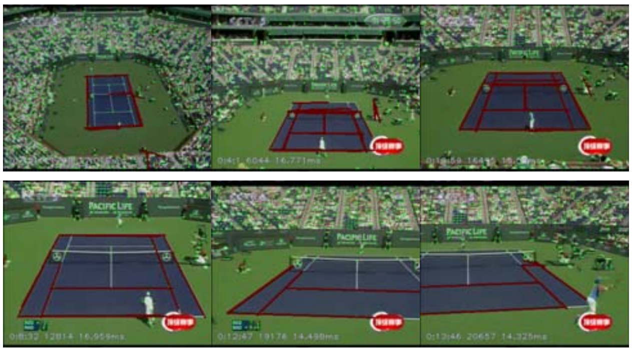


Figure 7: Examples at different camera view

We compare proposed algorithm with Standard Hough transform (SHT), Progressive Probabilistic Hough transform (PPHT). Running time of algorithm is described in table 1. The proposed algorithm is much faster than others.

TABLE I. AVERAGE RUNNING TIME

	Our	**SHT**	**PPHT**
P4 2.9G	11.8	35.3	49.4
Core 2	8.74	14.4	21.8

The demo video and binary code can be downloaded from: http://www.shenlejun.cn/download/09line.zip.

IV. CONCLUSIONS AND FUTURE WORK

This paper has presented a fast and accurate grid-like target detector in camera calibration.

We believe that our algorithm is a good framework to detect grid-like target not only for sport video but also for many other applications if they have the property: "straight lines link together".

The proposed algorithm has solved the "data missing" problem when the camera zooming or panning. But the extend line scan has not yet perfectly fix this problem. In the future we will seek for more robust methods.

ACKNOWLEDGMENT

This work is supported by the Natural Science Foundation of Sichuan Education Commission of China (No. 08ZC043) and China Sport Science Society.

[1] J. Assfalg, M. Bertini, C. Colombo and A. Del Bimbo, "Semantic annotation of sports videos," IEEE Multimedia, vol. 9, pp. 52-60, 2002.

[2] X. Yu, N.J. Jiang, L.F. Cheong, H.W. Leong, and X. Yan, "Automatic camera calibration of broadcast tennis video with applications to 3D virtual content insertion and ball detection and tracking," Journal of Computer Vision and Image Understanding, vol. 113, pp. 643-652, 2009.

[3] WANG Lan SHEN Le-jun KE Zun-yu. "RANSAC based visual measurement of point on ground plane and application in sport," Journal of China Sport Science And Technology, vol. 44, pp. 12-16, 2008.

[4] X Yu, HC Lai, SXF Liu, HW Leong, "A gridding Hough transform for detecting the straight lines in sports video," IEEE Conf. on Multimedia and Expo(ICME), vol. pp. , 2005.

[5] D. Farin, S. Krabbe, P. H. N. de With, and W. Effelsberg, "Robust camera calibration for sport videos using court models," In SPIE Storage and Retrieval Methods and Applications for Multimedia, vol. 5307, pp. 80-91,2004.

[6] D. Chen, Y. Wang, and G. Zhang, "A new sub-pixel detector for grid target points in camera calibration," The 20th Congress of International Commission for Optics, vol 6027, pp. 675-680, 2006.

[7] R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision," Cambridge University Press, 2000.

[8] C. Galambos, J. Kittler, and J. Matas, "Progressive probabilistic hough transform for line detection," IEEE International Conf. on Computer Vision and Pattern Recognition, vol. 1, pp. 1554, 1999.

[9] Harris, C., Stephens, M, "A combined corner and edge detector," In Proceedings of the Alvey Vision Conference, pp. 147-151, 1988.

[10] Michael J. Jones, James M. Rehg, "Statistical Color Models with Application to Skin Detection," IEEE International Conf. on Computer Vision and Pattern Recognition, vol. 1, pp.1274, 1999.

[11] C. Galambos, J. Kittler, and J. Matas, "Progressive probabilistic hough transform for line detection," IEEE International Conf. on Computer Vision and Pattern Recognition, vol.1, pp. 1554, 1999.