

Some Aspects of Pseudo Random Binary Array-Based Surface Characterization

H. J. W. Spoelder, F. M. Vos, Emil M. Petriu, *Senior Member, IEEE*, and F. C. A. Groen

Abstract—In this paper, we report on an algorithm which we have developed to model three-dimensional (3-D) scenes with color-coded Pseudo Random Binary Arrays (PRBAs). The algorithm comprises three steps: first, a graph recovery, then an iterative reinspection, and finally application of the PRBA information. The combination of these three steps allows the algorithm to become auto adaptive to the surface studied. This approach also opens ways to dynamically adapt a PRBA to the surface studied in order to achieve optimal resolution.

Index Terms—Biomedical signal analysis, image processing, optical measurements, pseudo noise processes.

I. INTRODUCTION

A BINARY n -stage shift register will produce a sequence which will repeat itself after at most 2^n elements [1]. Sequences of the maximum length are called pseudo random binary sequences (PRBSs). Pseudo Random Binary Arrays (PRBAs) are their two-dimensional (2-D) analogon. The binary information of both sequences and arrays can be coded in a number of ways depending on the application field. Color-coded PRBAs [1]–[6] can conveniently be used in the modeling of three-dimensional (3-D) scenes. To achieve this, an image in which a PRBA has been encoded is projected onto or reflected by the scene under investigation and subsequently recorded by a CCD camera. The major advantage of the use of PRBAs lies in their so-called windowing property: from a given minimal size, each subpattern of a PRBA is unique. This information allows for absolute positioning of the pattern which adds to potential robustness of the method.

In previous papers [4], [7], basic aspects of the application of this PRBA-based technique to the measurement of the shape of the human cornea have been discussed. We summarize here the major steps:

- Step 1) A PRBA is geometrically/color encoded into a 2-D pattern that is mirrored to the cornea.
- Step 2) In the reflected image uniquely characterized positions are localized.

Manuscript received May 18, 1998; revised September 28, 2000.

H. J. W. Spoelder is with the Faculty of Physics and Astronomy, Vrije Universiteit, 1081 HV Amsterdam, The Netherlands.

F. M. Vos is with the Faculty of Physics and Astronomy, Vrije Universiteit, 1081 HV Amsterdam, The Netherlands. He is also with the Faculty of Medicine, Vrije Universiteit, 1081 HV Amsterdam, The Netherlands.

E. M. Petriu is with the Department of Electrical Engineering, University of Ottawa, Ottawa, ON, Canada K1N 6N5.

F. C. A. Groen is with the Faculty of Mathematics, Computer Science, Physics and Astronomy, Universiteit van Amsterdam, 1098 SJ Amsterdam, The Netherlands.

Publisher Item Identifier S 0018-9456(00)10949-0.

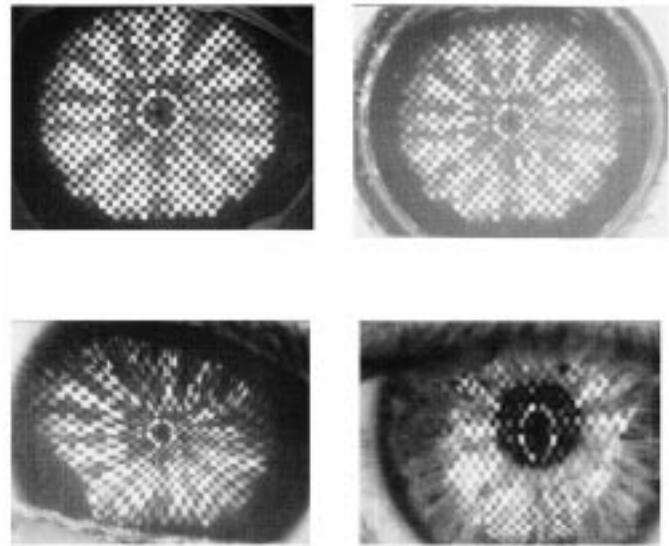


Fig. 1. Four test images studied in this paper: 1) regular sphere; 2) bisphere with smaller radius of curvature in the center than in the periphery; and 3) and 4) two severely deformed corneas.

- Step 3) A graph is built, connecting these positions into an ordered set.
- Step 4) Based on the PRBA information each position is unambiguously identified.
- Step 5) The positions thus defined are used to model the surface measured, e.g., the human cornea.

The successful application of this technique crucially depends on the robustness with which Steps 2, 3, and 4 can be performed.

For smooth surfaces these steps are easy to perform. However, when the surface studied has highly irregular features severe, local, distortions of the reflected image will be induced [see Fig. 1(3) and (4)]. Reliable graph reconstruction (Step 3) is then problematic.

In this paper, we present an algorithm which allows robust and fast graph reconstruction. We will illustrate the algorithm to a set of four test images which are representative for our application. These four cases are 1) a steel sphere, 2) a steel bisphere with smaller radius of curvature in the center than in the periphery, and 3) and 4) two severely deformed corneas. The four images which serve as input for Step 2 are shown in Fig. 1. Especially case 3) and 4) are typical examples of the earlier mentioned high irregular surfaces. The basic principles behind the algorithm are readily applied in other application fields.

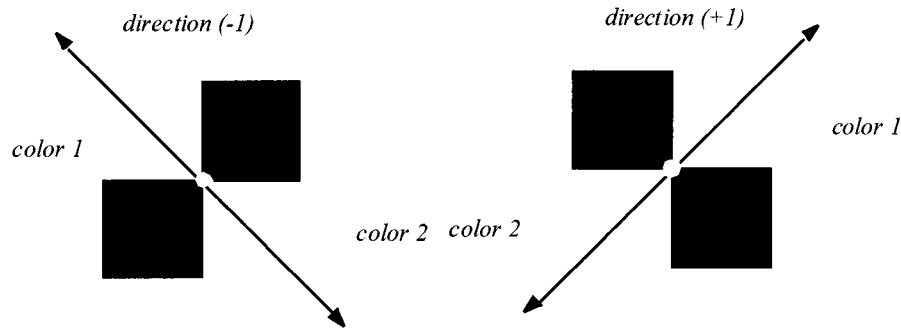


Fig. 2. Graphical representation of the attributes of a crossing: the position, two colors, and a color direction.

II. BACKGROUND AND DEFINITIONS

For the analysis of the images we use follow the following procedure (see also [8]). First we apply a matched filter of fixed size to the grey intensity image which has been obtained by summing pixelwise the r , g , b values of the color input image; the size of the matched filter (in our case 7×7) is determined by the size (in pixels) of the undistorted basic pattern. The positions looked for are characterized by a maximum absolute response to the matched filter. False positive identifications are reduced by only accepting those locations with responses that exceed a preset threshold T_0 , at the expense of introducing false negatives. For subpixel localization we follow again [8].

In the input image three colors are present: color_1 and color_2, which code the PRBA information, and black which allows for separation of the information of the PRBA bit pattern (for details see [4]). Application of a K-means clustering algorithm enables a tentative assignment of the colors involved.

All in all, per unique position, the following information is present: a sub-pixel localization (in image coordinates), a (binary) direction sense, and a tentative color assignment. This is illustrated in Fig. 2.

Thus, the image contains three information sources: 1) geometrical, 2) color, and 3) PRBA. Simple nearest neighbor algorithms exploit only the first source. However, for the irregular surfaces, like pathological corneas, this is not sufficient to guarantee the reconstruction of a consistent graph. Proper use of all information present allows for the definition of a far more robust algorithm, which we present in the following sections. The first step, referred to as basic algorithm, exploits simultaneous geometrical and color information. This results in the construction of a (small) consistent graph that serves as starting point for the next step. In this step, reinspection, the geometrical continuity of graph constructed in the first phase is exploited for further extension of the graph. Finally in the last step, the PRBA information is used to validate and position the resulting graph.

III. BASIC ALGORITHM

The starting point of the algorithm is an unordered set of points which have a directional sense of color. The data are imperfect due to a number of reasons:

- 1) false positives and/or false negatives,
- 2) possibly erroneous color assignments,
- 3) possible erroneous localizations.

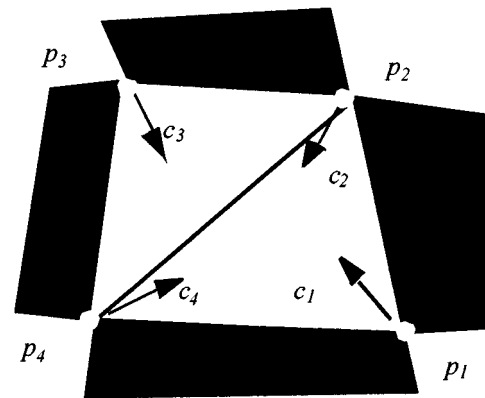


Fig. 3. Triangle built from positions p_1 , p_2 , p_4 is accepted only if $c_1 == c_2 == c_4$ (compare with Fig. 2); similarly triangle p_2 , p_3 , p_4 is accepted if $c_2 == c_3 == c_4$.

The first task is to regroup these points into an ordered graph taking into account these possible disturbances. From Fig. 1, it is furthermore clear that local variations in the deformation of the input image may be as large as a factor of three which makes the use of a fixed size matched filter questionable.

To meet with all these demands we propose the following strategy:

- 1) A Voronoi triangulation of the detected positions, which covers the point space by maximizing the minimal angle of the triangles, is taken as a starting point. The triangles defined are validated using the color information by imposing the constraint that the three corner points agree on the color of the interior (see Fig. 3). Thus this phase results in the creation of a set of triangles with a color attribute (black/color_1/color_2).
- 2) Next triangles that share a common line and have the same color are merged into "quadrilaterals." Any remaining isolated triangle is withdrawn from further processing.
- 3) Subsequently quadrilaterals that only have neighbors at top and bottom or on the left and the right are rejected. With this we aim to remove structures as indicated in Fig. 4(a) that skip a row or a column. It should be clear that with this criterion no image features are thrown away as the corner points involved are still part of nonrejected quadrilaterals.
- 4) Those combinations of colored and black quadrilaterals that have only one point in common are removed. This is illustrated in Fig. 4(b).

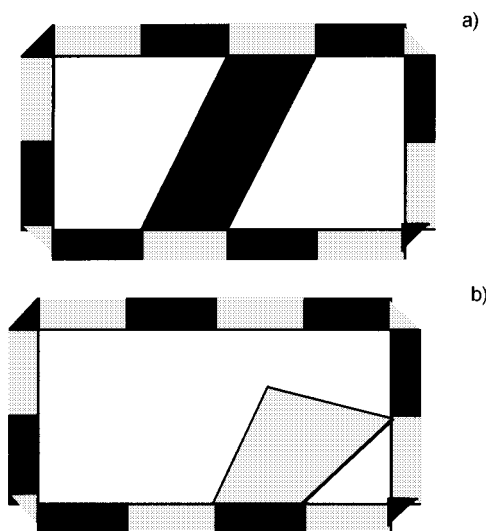


Fig. 4. Examples of graph fragments: the black structures represent black quadrilaterals, the grey represent colored quadrilaterals. Case (a) is removed by applying Step 3 whereas case (b) is removed by Step 4 of the algorithm.

TABLE I
NUMBER OF GRAPH STRUCTURES (TRIANGELS/QUADRILATERALS) REJECTED AT EACH PHASE OF THE RECOVERY FOR EACH OF THE IMAGES IN FIG. 1

Criterion	Input Image Number			
	1	2	3	4
1	23	17	31	29
2	117	186	281	266
3	0	1	7	6
4	12	40	92	78

Application of these four criteria yields in a per position assignment of a nearest neighbor in four directions: left, right, up, and down. The order in which criteria 3) and 4) are applied is irrelevant since they are orthogonal.

- 5) Finally the graph is reconstructed by traversing the two sets of quadrilaterals, black and colored, in a strictly alternating way using the nearest neighbor information.

To measure the relative importance of these criteria we have processed the four images shown in Section I. Table I summarizes the effect of the four steps by giving the number of triangles/quadrilaterals rejected. To put this number in perspective it should be noted that the maximum number of quadrilaterals in each of the images is 712. From Table I, it can be concluded that, as there are no all zero entries for any image, all criteria contribute. Step 2 clearly represents the largest reduction in the graph structure.

As the deformation of the input image increases (going from image 1 to 4) it appears that overall more graph structures are rejected. The graphs reconstructed for the test images are shown in Fig. 5.

These results show that despite severe malformations consistent graphs are built at the expense of the introduction of false negatives. Especially the boundaries of test images 3) and 4) show the presence of false positives.

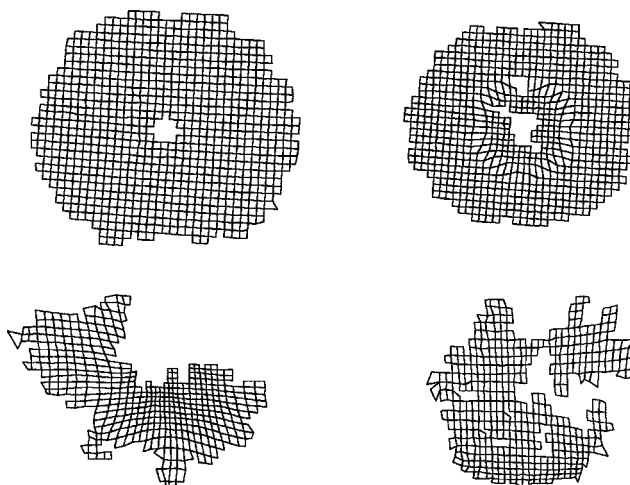


Fig. 5. Graphs reconstructed from the input images of Fig. 1 after application of the basic algorithm.

IV. ALGORITHMIC EXTENSION 1: PATTERN REINSPECTION

Visual comparison between the input images (Fig. 1) and the graphs reconstructed with the basic algorithm (Fig. 5) suggest that a further improvement may still be possible, especially in the severely deformed cases 3) and 4).

This improvement is based on a reinspection of the input image, driven by the already recovered graph. Extrapolation of the (grid lines of the) graph is used to define a search area for false negatives and detect false positives. These false negatives may result from a failure to meet the present threshold value T_0 introduced in Section II.

The reinspection basically exploits the principle of continuity of the geometrical shape. Consider the case shown in Fig. 6. In the basic graph c_1 has no right-hand side neighbor which makes this side eligible for reinspection. The line from c_0 to c_1 in the figure presents a connection which has been made in the previous phase. Even with severe distortions we have noticed that the direction as well as the length of such edges change only moderately over the graph. Therefore, the center of a search area for c_2 is geometrically determined by extrapolating $c_0 - c_1$ by l in the direction of (c_0, c_1) . The shape of the area limits the acceptable deviation in length and direction of the edge. Based on experiments we use a square with sides $l/2$. Note that implicitly this limits the frequency of the changes in length/direction that can be detected and thus represents *a priori* knowledge on the object studied. The search area becomes more complicated when for instance c_3 , the right-hand-side neighbor of c_2 , exists. In such cases we take the convex hull of all search areas defined.

The complete outer edge of the boundary quadrilaterals are always subject to a reinspection.

Within this search area we are reapplying the matched filter to the input image. The size of the matched filter is taken to be the uneven number which is closest to l (expressed in pixels). Note that we now adapt the size of the matched filter to the size of the pattern.

As before, the location of the new crossing is defined by the local extreme of the response of the matched filter. Additionally it is required that the average responses at discrete distances

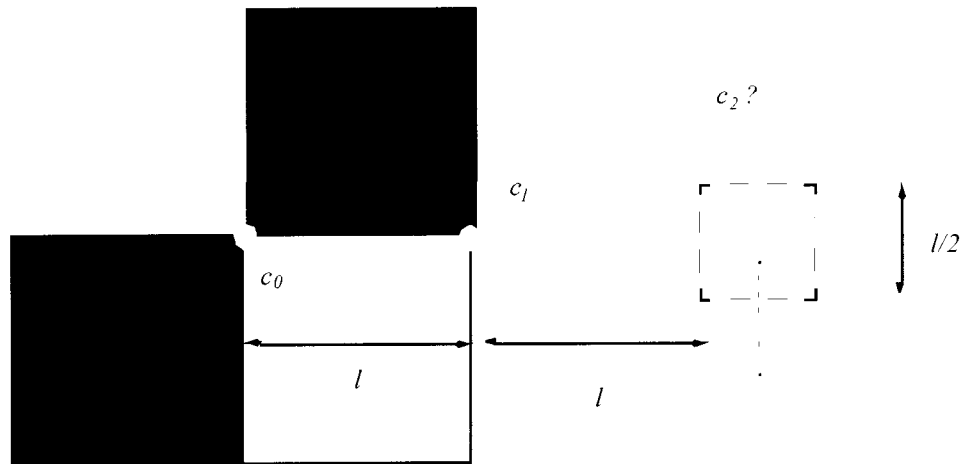


Fig. 6. Definition of the search area in which a false positive may be located.

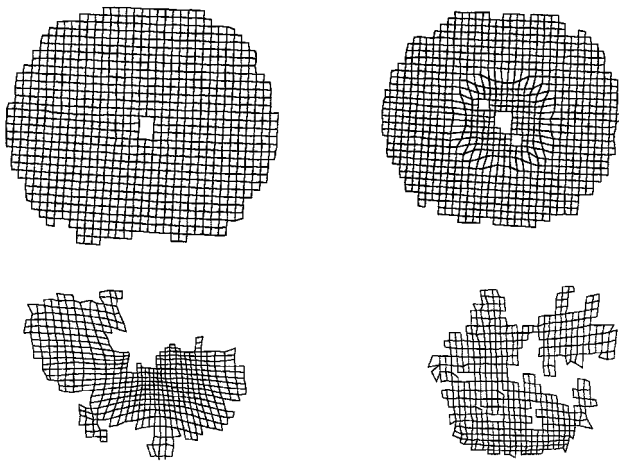


Fig. 7. After reinspection, the visible pattern is almost fully recovered (compare with Fig. 5).

from the extreme (i.e., $[1, \sqrt{2}, \sqrt{2}, \sqrt{5}, \sqrt{8}]$ for a 3×3 environment) form a monotonically decreasing sequence. By applying this criterion, we aim to remove any false positives. Since this criterion also implicitly thresholds the extreme, the threshold criterion (T_0) is no longer applied in this phase. A tentative point is accepted when it meets this criterion.

The process can be applied iteratively until no more tentative points can be found, i.e., the complete graph has been recovered, or each tentative point fails to meet the acceptance test. Applying this reinspection to the test images produces the graphs shown in Fig. 7. Table II collates the absolute number of quadrilaterals additionally recovered. Visually and numerically the improvement of the graph recovered is evident. For both images the iteration converges in six steps. To test the consistency and the accuracy of the pattern reinspection, the following experiment was done. From positions recovered from image 1 we selected a limited set of (initial) points from the middle part of the image [see Fig. 8(1)]. Subsequently, we executed and monitored the pattern reinspection. Visually this process is shown in Fig. 8(2)–(5). To evaluate the accuracy, we calculated the root mean square distance between the initially localized crossing of the basic algorithm and those found with the method outline

TABLE II
NUMBER OF QUADRILATERALS GAINED BY REINSPECTING IMAGES (3) AND (4). ITERATION 0 DENOTES THE STARTING NUMBER AFTER THE BASIC ALGORITHM IS APPLIED

Iteration	Image 3	Image 4
0	(490)	(482)
1	+89	+98
2	+18	+37
3	+10	+12
4	+9	+5
5	+3	+2
6	+0	+0

above. This average distance was found to be 0.1 pixel. This result is not significantly different from the standard localization error of 0.05 pixel ([7]). One might expect a spatial dependency of this localization error due to accumulative effects. To investigate this we have plotted the spatial distribution of the residues which show no such dependency. Therefore, both methods can be regarded as equally accurate.

V. ALGORITHMIC EXTENSION 2: PRBA VALIDATION

Having reconstructed the graph it is now possible to extract the PRBA information from the input image registered and match it against the stimulus. Aspects of this matching have been discussed in detail elsewhere [4]–[8]. It suffices here to state that a match can only be made when the PRBA recovered possesses a minimal size.

For our typical application and PRBA used this size is 12 bits (for noise free situations) and approximately 100 bits (for situation where 20% of colors is misinterpreted). Therefore prior to the PRBA matching, graphs recovered smaller than this threshold value, are removed. The PRBA validation allows for detection of complicated errors which are caused by complex shapes under investigation and cannot be detected by the geometrical tests given above.

However, for the above application these errors can be excluded. Therefore the final phase of the algorithm is the positional relocation of the graphs reconstructed.

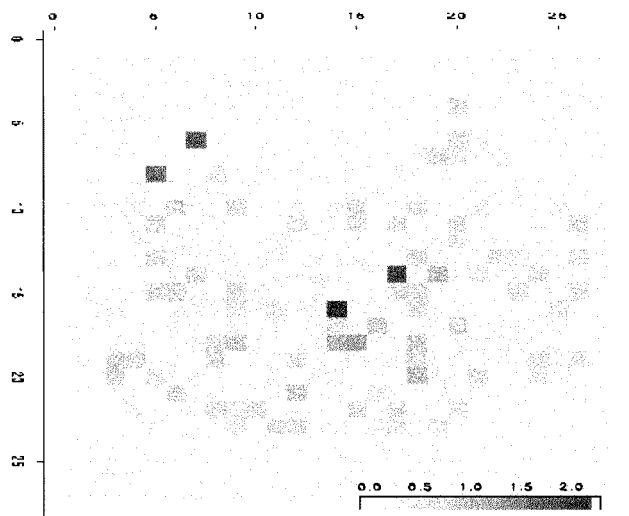


Fig. 8. Spatial distribution of difference in relocation between standard algorithm and dilation experiment for data from image (1). Grey scale runs from 0 (white) to 2 pixels (black).

VI. IMPLEMENTATIONAL ISSUES

The algorithm described in this paper is implemented in C and comprises about 3000 lines of code. The algorithm is centered around the manipulation of a list containing the information of the points of the graph. Fundamental is the list of points detected in the image. Derived from this list is an edge list and a triangle list (triple edge list). Finally each quadrilateral refers to two triangles.

In the implementation two separate quadrilateral lists are maintained, one for the colored and one for the black quadrilaterals. Each item contains a valid field indicating if the item has been rejected and if so for what reason. Efficiently looking up a line segment (Step 2 of Section I, or a point (Steps 3 and 4 of Section I) in these lists is all important in this algorithm. Straightforward implementation would yield $O(N^2)$ search algorithms (N being the number of points involved). For all practical purposes more efficient processing is required.

To realize this we use a hashing scheme by which we assign an unique number to each line segment. Let N be the total number of points and let c_1 and c_2 be the index of two points from this set, i.e. $c_0, c_1 \in [1, N]$. The hash value of line segment c_1, c_2 is then given by

$$S = 10^{\lceil \log N \rceil}$$

$$\text{Hash}(c_0, c_1) = \text{Min}(c_0, c_1) * S + \text{Max}(c_0, c_1)$$

Sorting this list (e.g., with quick sort) requires $\log N$ operations, while for searching for points and/or line segments, a binary search method on the hash value can be exploited. This again is a $\log N$ operation.

The total processing of the input image takes about 5 s on an IBM RS6000/43P (133 MHz) divided equally over the basic algorithm (searching) and the reinspection phase (matched filter). More advanced searching strategies and reuse of already computed matched filter convolutions will allow a further speed-up of the algorithm if necessary.

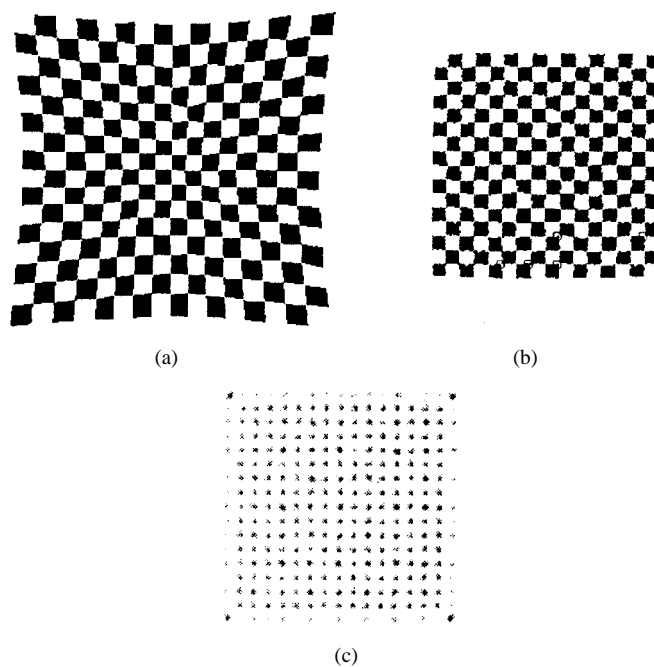


Fig. 9. Example of dynamic adaptation of a pattern. The input (a) is deformed in such a way that when registered by a sphere (b) it is uniformly spread, and (c) gives the search area for the matched filter and its response.

VII. DISCUSSION AND FUTURE WORK

The algorithm presented provides a robust way to analyze PRBA-based images for surface characterization. The images (3) and (4) could not be analyzed with a nearest neighbor algorithm. One of the attractive features of the three pass scheme is that it effectively makes the algorithm auto-adaptive to the size of the information of the input image. Reinspection increases the number of quadrilaterals recovered on the average by 25% allowing a more accurate surface determination.

The sole restriction in the current implementation is the size of the area used for reinspection of the image. This can be avoided if the second phase is not solely based on geometrical considerations but also exploits the PRBA information. With the aid of this the number of missing bits, i.e., positions, can be determined.

In our study the size of the undistorted quadrilaterals was 15×15 pixels (on the CCD). We found a size of 5×5 pixels to be a lower limit of the (distorted) quadrilaterals size for both reliable color characterization in the quadrilateral and localization of the cornerpoint. This span of a factor of 3 corresponds in the case of measurements on the human cornea with the ability to detect μm deviations on a sphere with a radius of 8 mm. When smaller deviations need to be detected a static regular input pattern will not be adequate. Then a dynamic pattern will have to be used to locally optimize the ratio between undistorted and distorted quadrilaterals. One of the possible feedback mechanisms is to spread as uniformly as possible the positions in the registered (output) image by deforming the input image given the irregular shape under investigation. It is clear that this facilitates the graph reconstruction procedure considerably and probes the surface most intensively where the changes are at a maximum.

Preliminary simulation studies have shown that this principle of the pattern driven matched filter convolution is applicable in

such cases and greatly reduces the amount of computational time needed. An example of this is shown in Fig. 9. Here, for simplicity sake, we assume a sphere as mirroring surface. Starting from an uniform input image [Fig. 9(a)] the resulting, output, image [Fig. 9(b)] is calculated such that when mirrored to a sphere it gives back an uniform pattern. Fig. 9(c) gives the areas in which the matched filter is applied. For a 300×300 image containing 16×16 black/white squares the computational time is 3 s. However, since this is done with a prototype version of software, significant speedups will be achievable.

In conclusion, this approach opens new possibilities for optical measurements of surfaces.

ACKNOWLEDGMENT

The authors wish to acknowledge the assistance of C. Bakker and D. Germans in the various stages of the development and coding of the algorithm.

REFERENCES

- [1] F. J. MacWilliams and N. J. A. Sloane, "Pseudo-random sequence and arrays," *Proc. IEEE*, vol. 64, pp. 1715–1729, 1976.
- [2] G. Hu and G. Stockman, "3-D surface solution using structured light and constraint propagation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 4, pp. 390–402, 1989.
- [3] G. Wade, *Signal Coding and Processing*. Cambridge, MA: Cambridge Univ. Press, 1994.
- [4] F. M. Vos, G. L. van der Heijde, H. J. W. Spoelder, I. H. M. van Stokkum, and F. C. A. Groen, "A new PRBA-based instrument to measure the shape of the cornea," *IEEE Trans. Instrum. Meas.*, vol. 46, pp. 794–797, Aug. 1997.
- [5] E. Petriu, "Absolute type position transducers using a pseudorandom encoding," *IEEE Trans. Instrum. Meas.*, vol. 36, pp. 950–955, Aug. 1987.
- [6] K. L. Boyer and A. C. Kak, "Color-encoded structured light for rapid active ranging," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 14–27, 1987.
- [7] H. J. W. Spoelder, F. M. Vos, E. M. Petriu, and F. C. A. Groen, "A study of the robustness of Pseudo Random Binary Array based surface characterization," in *Proc. IMTC'97*, Ottawa, ON, Canada, 1997, pp. 830–835.
- [8] P. Vuylsteke and A. Oosterlinck, "Range image acquisition with a single binary-encoded light pattern," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 148–164, Feb. 1990.
- [9] V. Srinivasan, L. R. Nackman, J. M. Tang, and S. N. Meshkat, "Automatic mesh generation using the symmetric axis transform of polygonal domains," *Proc. IEEE*, vol. 80, pp. 1485–1501, Sept. 1992.

H. J. W. Spoelder, photograph and biography not available at the time of publication.

F. M. Vos, photograph and biography not available at the time of publication.

Emil M. Petriu (M'86–SM'88) is a Professor of electrical and computer engineering and Director of the School of Information Technology and Engineering (SITE), University of Ottawa, Ottawa, ON, Canada, where he has been since 1985. His research interests include test and measurement systems, interactive virtual environments, robot sensing and perception, neural-networks, and control. He has published more than 120 technical papers, authored two books, edited two other books, and received two patents.

Dr. Petriu is a Fellow of the Engineering Institute of Canada. He is Vice President (Publications), member of the AdCom, and Co-Chair of TC-15, IEEE Instrumentation and Measurement Society. He is an Associate Editor of the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT and member of the Editorial Board of *IEEE I&M Magazine*. He is a Registered Professional Engineer.

F. C. A. Groen, photograph and biography not available at the time of publication.