# WIRE PATTERN AND COGNITIVE GRAPH REPRESENTATION OF TEXTURED IMAGES

Yehoshua Y. Zeevi and Oliver A. Hilsenrath

Department of Electrical Engineering
Technion - Israel Institute of Technology
Haifa 32000, Israel

## ABSTRACT

A new approach is presented to extract wire-pattern contours out of textured images, by means of a successive scanning algorithm which is applied to noisy raw data. Each frame is scanned in a different orientation and the resultant pattern is compared with the previous one. A size threshold is determined, based on a priori information regarding object size distribution, and closed contours of size below the threshold are successively eliminated. The end result is a pattern of the most significant contours.

A neighbourhood region-interaction map, which is the dual of the contour graph, is generated as a byproduct of the process. This interaction map provides some of the most important information required for the execution of various scene understanding strategies.

## 1. INTRODUCTION

Logic reasoning has been successfully applied over the last twenty years in the analysis of labeled graphs reduced from textured images and efficient algorithms were developed for processing and understanding of such a synthetic image representation [1]. However, the problem of extracting such a representation still remains outstanding.

We present a new approach to contour/wire pattern extraction, wherein the preprocessing and "understanding" stages are interconnected at a certain extent. The object of the study is to obtain Real World Graph (RWG) representations of both polyhedral and smooth object two-dimensional projections [2]. The RWG is then mapped onto a dual-cognitive graph in which environment the RWG is separated into subgraphs corresponding to the individual objects. In the case of a polyhedral scene, the RWG contains the visible edges and vertices of individual objects as well as those generated by occlusion of objects. In the case of a smooth world scene, the resultant graph contains the silhouette edges of individual objects and of their occlusions, as well as the edges generated by $m$ intensity level-crossings [3].

## 2. PREPROCESSING

The first stage of the process is application of an arbitrary low threshold edge detector on the original image (Fig. 1(b)). The result is noisy and non-optimal but, as it will be emphasized, false alarm in the present case is preferable than non-detection of edges. However a median filter is applied to mainly eliminating isolated small size blobs.

## 3. COLORING BY A ROTATING RASTER

Next, a coloring algorithm is applied as follows: An arbitrarily directed raster is scanning the edge detected image modifying the color of the scanpath whenever an edge is crossed. The next color is arbitrarily selected, but is identical to the color employed in the upper-previous path, if no contour fluctuation has occurred with respect to the upper-previous path.

The result of a first scan, from left-to-right, is shown in Fig. 1(c). The different color strips within a same contour are due to fluctuations in the contours or contour crossings.

Actually each color is supposed to resemble a closed contour and, obviously, this is not obtained in a single scan.

A second scan is employed with a different orientation and with the same logic. Clearly a previously uniquely colored region will not be cromatically partitioned due to new contour fluctuations (Fig. 1(d)). Continuing the coloring in a few more orientations (Fig. 1(e)-(h)) the process clearly converges to a unique color for each closed contour.

## 4. MOST SIGNIFICANT CONTOURS AND NEIGHBOURHOOD INTERACTIONS

Starting from the high sensitivity edge detected scene in Fig. 1(b), the probability of spurious contours, creating even "noisy" closed contours, is substantial. The color-histogram in Fig. 2(a), containing the color of the closed contour on the x axis versus the number of pixels in each region on the y axis, illustrates the size of the various regions (the largest region is the background). The spurious regions are eliminated by thresholding the color-histogram and merging the areas, the size of which are below the threshold, with one of the neighbors (i.e. merging colors).

An important byproduct of the above process is the knowledge obtained during scanning, of which colors interact with one another – i.e. which regions are neighbours. Looking at the contours as a graph, the colors represent facets of the graph and the color interconnection matrix in Fig. 2(b) represents the interconnections of the dual graph. This is the major input for scene understanding processes [1,2,3].

The results of two different images are treated by the above methodology in Fig. 3, this time containing also a polyhedral scene. The resulting facet interconnection matrix is nothing but the neighbourhood relation matrix of the facets within and between the polyhedra [2].

## 5. CONCLUSIONS

The algorithm of sequential scanning and coloring along different orientations has been found useful with regard to: edge detection, edge extraction, closed contour delineation, spurious region elimination and the definition of neighbourhood relation between closed contours. The latter is in particular crucial in 3D analysis using the cognitive graph approach [2], but it is just as important in other areas of image analysis and machine vision.

The algorithm is shown to be efficient for convex and simple concave structures in scenaries common to indoors robotic vision systems.

## ACKNOWLEDGEMENT

The research has been supported in part by the TECHNION FUND FOR PROMOTION OF RESEARCH, and by the TECHNION V.P. GRANT.

## REFERENCES

[1] Guzman, A. "Decomposition of a visual scene into three dimensional bodies". AFIPS - Vol. 33, Thompson Co. 1968.

[2] Hilsenrath, O.A., Zeevi, Y.Y. "3D Cuboid Scene Understanding by a Mixed Cognitive Graph and Log.-Complex Mapping Paradigm". Proc. 9th Int. Conf. on Pat. Recog. - Rome, Italy 1988 (to appear).

[3] Horn, B., Brooks, M. "The variational approach to shape from shading". Biol. Cybern. 53 1986.

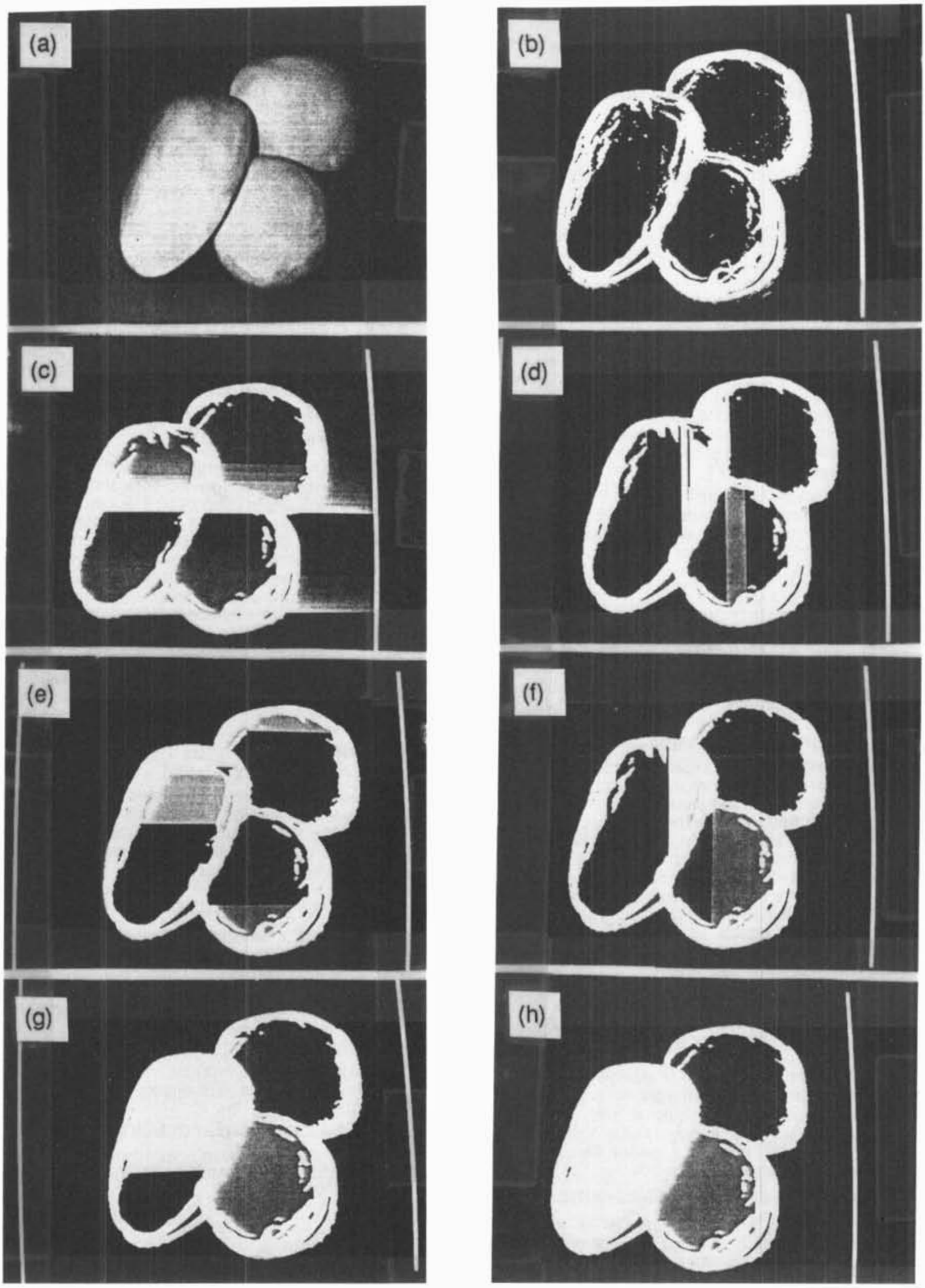



Fig. 1: The different stages of the pre-processing and the rotating raster coloring process operating on a smooth object scene: (a) the original image. (b) the edge detected result, (c)-(h) six coloring sessions with six raster orientations, with (h) representing the result.

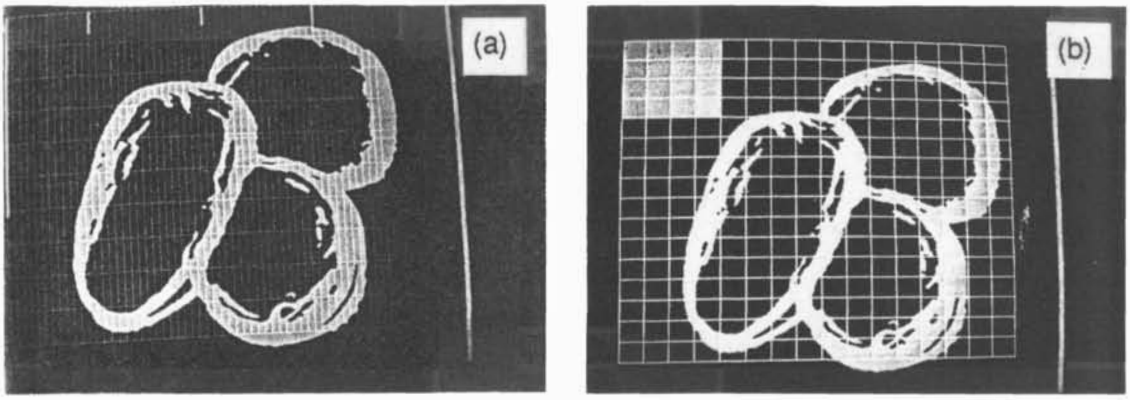
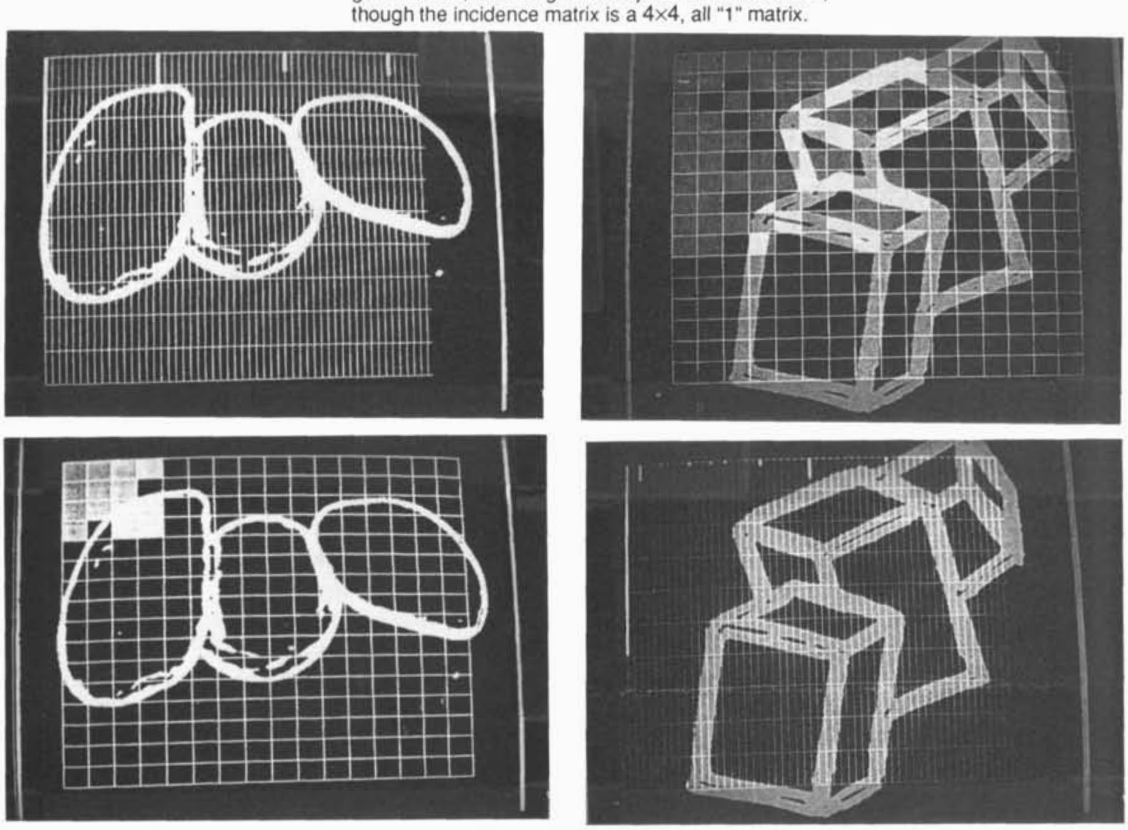



Fig. 2: (a) The color-histogram superimposed on the contour image. The x axis represents the region-color (actually the region-index) and the y axis represents the size of the region (number of pixels with the same color). Thus four regions were found: three objects and background (b) the color incidence matrix also superimposed on the contours. Each square represents the $a_{ij}$ coefficient between regions $i$ and $j$ (white is "1", black is "0"). In the given scene, each region is adjacent with the other, though the incidence matrix is a 4×4, all "1" matrix.



(a) (b)

Fig.3: The results of two other coloring sessions (color-histogram and neighbourhood matrix):
(a) another smooth object scene,
(b) a polyhedral scene (here the neighbourhood matrix represents the incidence matrix of the dual).