July 1989

Fast Image Segmentation

Peter I. Corke
University of Pennsylvania

Helen I. Anderson
University of Pennsylvania

Follow this and additional works at: https://repository.upenn.edu/cis_reports

Recommended Citation


This paper is posted at ScholarlyCommons. https://repository.upenn.edu/cis_reports/841
For more information, please contact repository@pobox.upenn.edu.
Fast Image Segmentation

Abstract
Image segmentation remains one of the greatest problems in machine vision. The technique described here takes an image and a geometric description of the object required, determines multiple binary thresholds to segment the image, and combines the information from the appropriate thresholds. By utilizing region-growing hardware it is possible to achieve segmentation in less than 2 seconds.

Comments
FAST IMAGE SEGMENTATION

Peter I. Corke' and
Helen I. Anderson

MS-CIS-89-46
GRASP LAB 185

Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104

July 1989

Acknowledgements: This work was in part supported by: Air Force grant AFOSR F49620-85-K-0018, ARPA grant N0014-88-K-0630, DARPA/ONR grant N0014-85-K-0807, NSF grant DCR-8410771, IRI84-10413-A02, NSF-CER/DCR82-19196 A02, NATO grant No.0224/85, NASA NAG5-1045, ONR SB-35923-0, Army/DAAG-29-84-K-0061, DAA29-84-9-0027, NIH grant NS-10939-11 as part of Cerebrovascular Research Center, NIH 1-RO1-NS-23636-01, NSF INT85-14199, NSF DMC85-17315, by DEC Corp., IBM Corp., LORD Corp. and E.I. du Pont Corp.
Fast Image Segmentation

Peter I. Corke
Helen L. Anderson

Computer and Information Science Department
University of Pennsylvania
Philadelphia, PA 19104

Abstract

Image segmentation remains one of the greatest problems in machine vision. The technique described here takes an image and a geometric description of the object required, determines multiple binary thresholds to segment the image, and combines the information from the appropriate thresholds. By utilizing region-growing hardware it is possible to achieve segmentation in less than 2 seconds.

1Research Scientist, CSIRO Division of Manufacturing Technology, Melbourne, Australia.

2Acknowledgement: This work was in part supported by: NSF grant DCR-8410771, Air Force grant AFOSR F49620-85-K-0018, Army/DAAG-29-84-K-0061, NSF-CER/DCR82-19196 Ao2, NASA NAG5-1045, ONR SB-35923-0, NSF INT85-14199, NSF DMC85-17315, ARPA N0014-88-K-0632, NATO grant No.0224/85. by DEC Corp., IBM Corp., LORD Corp. and E.I. du Pont Corp.
1 Introduction

Image segmentation remains one of the greatest problems in machine vision. Three major approaches are used today: simple thresholding for real-time implementations, complex algorithms in laboratory image processing systems such as grey-scale connectivity or histogram splitting, or knowledge-based systems which use substantial domain knowledge.

Firstly, for real-time and hardware implementations, simple pixel thresholding is used due to the small amount of computation required and the ease of binary image processing. However, the disadvantages of this scheme are numerous. In particular, choice of threshold, and robustness with changing and uneven lighting conditions cause great difficulty. An inherent problem in the approach is threshold selection. Many techniques for choosing a suitable threshold intensity have been developed [Weszka 78] [Sahoo 88]. Most are based on examination of the modalities of the intensity histogram. For a simple bimodal distribution this is relatively easy, but in reality most scenes are multimodal. [Rosenfeld 82] described a procedure using trial binary thresholds to select a single threshold which produces a desired property. However, any single threshold may not be suitable for a complex image.

Secondly, much literature has addressed the issue of segmentation of grey scale pixel data directly, using criteria such as local rates of intensity change [Haralick 84] or advanced thresholding techniques [Sahoo 88]. These techniques allow local control of threshold or border detection. Unfortunately, these algorithms are complex, time consuming and not well suited to real-time hardware implementation.

Thirdly, knowledge based algorithms such as [Riseman 87] and [Fua 87] use multiple complex techniques with substantial domain knowledge to produce segmentation and object identification. This type of technique is the most complex, and is not suitable for real-time image processing.

In this paper we present an extension of the first to segment a real scene based on the analysis of binary images obtained at different threshold levels. By using recently available hardware capable of performing connectivity analysis and moment-generation at video data rates, it is possible to perform numerous trial segmentations of a scene per second. The trial thresholds are chosen by a histogram of the whole image, such that thresholds which will produce a trivial segmentation (all white or black) are not used. Regions are chosen on the basis of shape analysis, independent of threshold value, using the output of the hardware region-growing unit. However, we extend this method to obtain regions from multiple thresholds.

The algorithms and software will be described in section 2, the hardware in section 3 and Appendix A, and the experiments in section 4.
2 Segmentation Algorithm

2.1 Shape factor

Control of threshold selection is exercised through measurement of region shape complexity, via a circularity factor. This method is used to select acceptable thresholds to produce regions with simple external shapes. This measurement has been described in [Rosenfeld 82]. The geometric shape factor, $\rho$, is defined as

$$\rho = \frac{4\pi \text{Area}}{\text{Perimeter}^2}$$

which has a maximum value of 1.0 for a circle, and lower values for shapes with more complex perimeters.

Perimeter is computed by an algorithm that examines a 3x3 window around each perimeter point and produces an appropriate perimeter length contribution depending upon the slope of the perimeter at that point. A lookup table of the perimeter contributions is included in the hardware. Experiments reveal a perimeter error of less than 2% with this scheme. Accurate perimeter estimation for objects with jagged edges is essential for accurate estimation of shape factor. To consider only exterior shape factors, an adjacency tree for the regions is created. Perimeters of child regions are subtracted and areas of child regions are added to region statistics. This method of perimeter determination approximates the perimeter length regardless of orientation by fitting a line to the boundary within the window, which is a technique described in [Ellis 79].

In experiments we have found that circles exhibit shape factors greater than 0.96, and that for irregularly shaped objects is invariant with respect to rotation. This indicates that perimeter is accurately determined.

Another difference in our approach is not attempting to find only one threshold to segment all objects of interest, which for many types of scene is impossible. Instead the objects can be considered as the union of regions matching the description at a number of different threshold levels.

2.2 The qualitative effects of threshold variation

As a first step we decided to examine qualitatively the effects of various threshold levels. The test scene consists of a tennis ball and some other objects lying on a textured carpet in the laboratory with no particular care given to lighting levels or evenness. Figure 2 shows the test scene and the thresholded scene at a number of different threshold levels. It can be seen that for very low or very high thresholds the result is uniform white or black respectively. At the right threshold the binary scene accurately reflects the test image. However at intermediate thresholds there is a considerable increase in small noise regions, merging of features in the image with each other.
and with the background. As threshold increases the lines on the ball become apparent, then with further increase the ball shrinks in area, and eventually disappears. A plot of the number of connected regions versus intensity is shown in Figure 3, and is perhaps more useful than traditional intensity histograms. The range of good thresholds is readily seen, and is delineated by spikes of high region count due to noise and other effects.

### 3 Image processing architecture

This approach is predicated on being able to rapidly analyze binary images so we use a newly available hardware unit capable of analyzing binary images at video data rates. The APA-512[APA 87](for Area Parameter Accelerator) is a hardware unit designed to accelerate the computation of area parameters of objects in a scene. The APA binarizes incoming video data and performs a single pass connectivity (simple-linkage region growing[Haralick 84]) analysis, and subsequent computation of moments up to second order, perimeter and bounding box. The APA performs very effective data reduction, reducing a 10Mpixel/s stream of grey-scale video data via a MAXBUS interface, to a stream of tokens representing objects in the scene, available via onboard shared memory. Appendix A provides additional details of APA operation.

A functional representation of the image processing subsystem is shown in Figure 1, and is based on Datacube pipeline processing modules.

The Datacube family of video processing modules are VMEbus boards that perform various operations on digital video data. The inter-module video data paths are patch cables installed by the user. The boards are controlled by a host computer via the VME bus. The video data paths run at 10Mpixels/s and are known as MAXBUS\(^3\). Horizontal and vertical timing is established by a separate timing bus linking all boards. All images are 512

\(^3\)Trademark of Datacube Inc.
Figure 2: Qualitative effect of threshold variation
Figure 3: Region count versus threshold from scene with high contrast

Figure 4: Histogram of region count versus threshold from scene in Figure 2
x 512 pixels, and all arithmetic is performed in 8 bits 2’s complement. The host processor in our case is a SUN workstation running Unix.

The original video stream is deinterlaced by a double-buffered framestore, offset by an intensity threshold \(\text{thresh}\), and translated by a lookup table which maps pixels to one of two grey levels, corresponding to the binary values black or white. Binary median filtering on a 3x3 neighbourhood is used to eliminate one or two pixel noise regions which may overwhelm the host processor. The threshold is controlled by the host processor. Additionally our experimental system allows canned images to be loaded into the framestore for processing by this algorithm.

3.1 Correcting for included regions

The APA area and perimeter parameters for each region are not directly usable in this experiment. The area parameter for a region is exclusive of all enclosed regions, while the perimeter parameter is total perimeter, both inner and outer. To generate total enclosed area and outer perimeter, a tree data structure is built from the APA topological information. The tree is traversed and corrected parameters are generated. The tree is then pruned to eliminate all regions whose area scale is less than 1% of the imaged area.

At each threshold level the reduced tree is examined to find objects with the appropriate shape factor, these are referred to as candidate objects. Each candidate is then compared against each element of an existing candidate list, and if not present in the list is appended. Each candidate in the list records the minimum and maximum intensity at which it is distinguishable. Candidates are compared to those in the list on the basis of approximate centroid location and size.

4 Experiments

In the following experiments the threshold selection strategy simply chooses 32 levels equally spaced between 0 and maximum, 127. This procedure is somewhat inefficient and could be assisted by knowledge of the intensity histogram which is readily computed by other Datacube hardware modules.

The program identifies all objects whose shape factor is greater than 0.7 (which corresponds to a rectangle with aspect ratio 2:1). It marks objects where \(\rho > 0.95\) as ellipses and those where \(0.7 < \rho < 0.95\) with rectangles. Figures 5, 6 and 7 show the ellipses and rectangles. In addition, they show the region number and the range of thresholds (in parentheses) over which the individual region is distinguishable. Note that the system ignores all regions which touch the edge of the processing window, which is indicated in the Figures, since the computed parameters for such regions will be incorrect.
4.1 Simple scene

This example, shown in Figure 5, contains a group of circular cookies, including one with a small chip, one with a large chip and one which is broken. Only cookies lying completely within the processing window are analyzed. The system finds all whole cookies. The broken cookie has $\rho < 0.7$ and is not marked. By adjusting the value of $\rho$ which is considered circular, the cookie with the larger chip can be identified.

4.2 Scene requiring Multiple Thresholds

This example, shown in Figure 6, is a CCD image of a synthetic scene generated by a PostScript program. No single threshold will segment all four circles in the scene, yet by combining information from different thresholds, all circles are found.

4.3 Complex scene

The scene in Figure 7 is a group of simple and complex objects against a textured background. It has objects of different shapes and uneven lighting. The system will segment only the simple shapes, and does not show the set of keys.

5 Future Directions and Conclusion

We have developed an adaptive segmentation system, based on the high speed analysis of binary images obtained at many different threshold levels. This system operates in approximately 2 seconds, and is able to do a better job of segmentation than any single binary threshold. It is faster than gray scale segmentations. It is able to segment all "simple" objects in the scene without any explicit user intervention or user-supplied parameters. This has been demonstrated on several, very different, types of scene.

The system could be expanded to utilize image intensity histograms, reducing the number of thresholds analyzed, and thus reducing processing time still further.

6 Acknowledgments

This research was conducted at the General Robotics and Sensory Perception (GRASP) Laboratory, University of Pennsylvania, and the support of Dr Ruzena Bajcsy, Chairman of the Computer and Information Science Department of the University of Pennsylvania and Director of that laboratory is acknowledged.
Circularity threshold 0.85

Circularity threshold 0.95

Figure 5: Simple scene - cookies
Figure 6: Scene requiring multiple thresholds

Figure 7: Complex scene
The authors are grateful to Vision Systems Ltd, Adelaide, Australia, for lending the APA-512 unit upon which this project is based. The first author’s visit to University of Pennsylvania was possible due to a CSIRO Overseas Fellowship.

A Appendix: APA-512

The APA-512[APA 87] is a hardware unit designed to accelerate the computation of area parameters of objects in a scene. It was conceived and prototyped by the CSIRO Division of Manufacturing Technology, Melbourne, Australia, in 1982-4, and is now manufactured by Vision Systems Ltd. of Adelaide, Australia.

The unit has some similarities to other hardware implementations of binary image processing systems. [Vuylsteke 81][Tropf 82][Froith 81] Andersson’s unit [Andersson 85][Andersson 88] computes moments of grey-scale data so as to improve accuracy when dealing with a quickly moving object. However it has no capability to detect and generate moments for multiple regions, and multiple regions will, if present, be merged into one moment set.

The APA binarizes incoming video data and performs a single pass connectivity (simple-linkage region growing) analysis. The connectivity unit commands a bank of 8 ALUs which update the region parameters (referred to as seeds stored in seed memory. The ALUs are implemented by custom gate arrays. The seed memory is dual ported to the host VMEbus so that seed parameters of completed regions may be read.

The APA performs very effective data reduction, reducing a 10Mpixel/s stream of grey-scale video data via a MAXBUS interface, to a stream of
tokens representing objects in the scene. The host processor screens the
tokens according to their parameters, and thus finds the objects of interest.

For each region the following parameters are computed:

- $\Sigma i$ number of pixels (zeroth moment)
- $\Sigma x$, $\Sigma y$ (first moments)
- $\Sigma x^2$, $\Sigma y^2$, $\Sigma xy$ (second moments)
- minimum and maximum x and y values for the region
- perimeter length
- a perimeter point
- region color (0 or 1)
- window edge contact

From these fundamental parameters, a number of commonly used area
parameters such as

- area
- centroid location
- circularity
- major and minor equivalent ellipse axis lengths
- object orientation (angle between major axis and horizontal)

may be calculated by the host processor. The perimeter point is the coordi-
nate of one pixel on the region's perimeter, and is used for those subsequent
operations that require traversal of the perimeter. The edge contact flag,
when set, indicates that the region touches the edge of the processing win-
dow and may be partially out of the image, in this case the parameters would
not represent the complete object.

Perimeter is computed by a sophisticated scheme that examines a 3x3
window around each perimeter point and produces an appropriate perime-
ter length contribution depending upon the slope of the perimeter at that
point. Experiments reveal a perimeter error of less than 2% with this scheme.
Accurate perimeter estimation for objects with jagged edges is essential for
accurate estimation of shape factor.

The APA-512 computes these parameters for each of up to 256 current
regions within the scene. Processing of the data is done in raster scan fashion,
and as the end of a region is detected the region label is placed in a queue and
the host is notified by an interrupt or a pollable status flag. The host may
read the region parameters and then return the region label to the APA for
reuse later in the frame, thus allowing processing of more than 256 objects within one frame. This feature is essential for processing non-trivial scenes which can contain several hundred regions of which only a few are of interest. Maximum processing time is one video frame time.

An additional feature of the APA is its ability to return region hierarchy information. When a region is complete the APA may be polled to recover the labels of already completed regions which were topologically contained within that region. This makes it possible to count the number of holes within an object, and compute the area of enclosed holes or internal perimeter.
References


