April 1988

Adaptive Image Segmentation

Helen L. Anderson
University of Pennsylvania

Ruzena Bajcsy
University of Pennsylvania

Max Mintz
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-88-26.

Adaptive Image Segmentation

Abstract

This paper introduces a general purpose scene segmentation system based on the model that the gradient value at region borders exceeds the gradient within regions. All internal and external parameters are identified and discussed, and the methods of selecting their values are specified. User-provided external parameters are based on segmentation scale: the approximate number of regions (within 50%) and typical perimeter:area ratio of objects of interest. Internal variables are assigned values adaptively, based on image data and the external parameters. The algorithm for region formation combines detected edges and a classical region growing procedure which is shown to perform better than either method alone. A confidence measure in the result is provided automatically, based on the match of the actual segmentation to the original model. Using this measure, there is confirmation whether or not the model and the external parameters are appropriate to the image data. This system is tested on many domains, including aerial photographs, small objects on plain and textured backgrounds, CT scans, stained brain tissue sections, white noise only and laser range images. The system is intended to be applied as one module in a larger vision system. The confidence measure provides a means to integrate the result of this segmentation and segmentations based on other modules. This system is also internally modular, so that another segmentation algorithm or another region formation algorithm could be included without redesigning the entire system.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-88-26.

Adaptive Image Segmentation

Helen L. Anderson[1] Ruzena Bajcsy [1]
Max Mintz[1]
Computer Science Department
University of Pennsylvania
Philadelphia, PA 19104

April 21, 1988

[1]Acknowledgement: This work was in part supported by: Air Force grant AFOSR F49620-85-K-0018, DARPA/ONR grant N0014-85-K-0807, NSF grant DCR-8410771, Army/DAAG-29-84-K-0061, NSF-CER/DCR82-19196 A02, NIH grant NS-10939-11 as part of Cerebrovascular Research Center, NIH 1-RO1-NS-23636-01, NSF INT85-14199, NSF DMC85-17315, ARPA N0014-85-K-0807, by DEC Corp., IBM Corp. and LORD Corp.

Abstract

This paper introduces a general purpose scene segmentation system based on the model that the gradient value at region borders exceeds the gradient within regions. All internal and external parameters are identified and discussed, and the methods of selecting their values are specified. User-provided external parameters are based on segmentation scale: the approximate number of regions (within 50%) and typical perimeter:area ratio of objects of interest. Internal variables are assigned values adaptively, based on image data and the external parameters. The algorithm for region formation combines detected edges and a classical region growing procedure which is shown to perform better than either method alone. A confidence measure in the result is provided automatically, based on the match of the actual segmentation to the original model. Using this measure, there is confirmation whether or not the model and the external parameters are appropriate to the image data. This system is tested on many domains, including aerial photographs, small objects on plain and textured backgrounds, CT scans, stained brain tissue sections, white noise only and laser range images. The system is intended to be applied as one module in a larger vision system. The confidence measure provides a means to integrate the result of this segmentation and segmentations based on other modules. This system is also internally modular, so that another segmentation algorithm or another region formation algorithm could be included without redesigning the entire system.

1 Introduction

Segmentation of a scene as a problem in Computer Vision has been studied with unreliability and instability remain in the segmentation of unfamiliar images without significant manual intervention. Models and structured programming concepts are being used in this research to bring order to the creative (and previously often chaotic) process of translating (sometimes vague) theoretical approaches to the machine vision process. This work is an attempt to produce a modular segmentation process that can be understood by others and changed without starting over at the beginning.

The task of segmentation is the grouping of pixels into equivalence classes with respect to some function. The problem is inherently underconstrained, non-deterministic and data-driven. We seek constraints which are general enough that they can be applied to a large class of problems, but which constrain the problem as much as possible. Program modularity, a primary characteristic of structured programs, requires a well-defined interface and self-contained logic flow. We define positionally invariant generic constraints (implemented as user-provided interface parameters) for the image segmentation process, as well as the logic flow for basic modules. The global segmentation process is based on explicit models, which allows prediction of the outcome of the process. In addition, measures are provided which confirm that the result is appropriate to the given image data.

The user interface parameters which we have found to be generic are based on the size and detail level of meaningful objects, which we refer to as the segmentation scale appropriate to the application. This is implemented as integral geometry (range of sizes of expected regions and the number of expected regions), topology, and very gross shape geometry (determined by the perimeter:area parameter). A global model of this type is flexible and does not make any assumptions on spatial arrangement of objects.

This theory is implemented and tested to understand how far one can take the segmentation process **without** *a priori* semantic (contextual) knowledge. We use only the image data and segmentation scale parameters. In order to understand when one must bring in semantic knowledge, the limits of the (bottom-up) modules must be explored. The limits of the modules limit the ability of higher level processes to deal with unexpected objects. In addition, this understanding leads to the ability to stop extensive processing of images containing no useful information.

For the discussion here, an algorithm is considered to be unstable if it can **only** work on a different image from the one (or few) for which it was designed if parameter values are reset by manual intervention. Instability can be overcome by adaptively setting values of variables within the modules. Adaptation is done by rating trial partitions with respect to a partition evaluation function. After the values are interated, the best partition, with respect to the partition value function and within range of the external (user) parameters, is chosen.

Unreliability occurs when an algorithm cannot work on a new image because the image data is not appropriate for the model which led to the algorithm's design. Unreliability can be overcome when the appropriateness of an algorithm can **automatically** be assessed. In our implementation, a measure of confidence in the result is provided, based on the match of the segmentation to the algorithm model, which will be described in Section 2.1. If other segmentation algorithms were also applied to the image, each measuring confidence in the data/model match, there would be a theoretical basis for automatic selection of the most appropriate image segmentation for the scene.

Thus this entire segmentation process can be used as a module in a more complete vision system. The user interface parameters are the only application dependent terms. Other self-adapting segmentation modules can be integrated by model-match confidence measures. We note that the user, which provides the parameters and uses the result, can be a person or another module.

There are several potential applications of the model-match confidence which are not fully developed in this work. First, in a given domain, if a particular adaptive segmentation system is known to produce good results, then the confidence measure can be applied to test whether any useful information is present without human supervision. Second, the confidence measure, which is applied globally here, can be applied locally to individual regions. This can be applied to adjust the width of an edge detector, adaptively and locally, for multi-scale edge detection.

2 The Segmentation Process

2.1 Global Parameters

All of the external parameters, the internal variables and the modules can be described as an n-tuple which produces a segmentation. It is assumed that the external parameters are known approximately from domain knowledge. Using a model of regions described below, internal modules and parameters have been selected.

To govern the entire process, the global model that regions and gradient edges are complementary is used. This was recommended as an approach by [Haralick 85] in his survey of segmentation algorithms and described by [Bajcsy 86]. The complementary relationship of regions and edges was also used to confirm boundaries produced by split-and-merge techniques in [Pavlidis 88]. Regions and edges were used to generate and verify rectilinear shape models in [Fua 87]. Integration of region-based and line-based operators has been implemented by [Riseman 87]. Since the use of lines for segmentation is based on the assumption that the objects to be segmented have straight borders or oriented texture patterns, Riseman *et al.* have done the integration of region and line information occur after low-level processing. In this way errors which occur in one process do not effect another process. However, edges (of the appropriate scale) can follow border contours, whether they are straight or not. Other segmentation models have been used, such as the maximum *a posteriori* model [Chou 87] [Cohen 87].

After the user specifies external parameters, values of internal variables are set automatically. [Beveridge 87] also sets internal variable values, using 5 pre-determined sensitivities and pre-calculated internal thresholds. However, for the variables used in this system, a much greater variety of image data can be handled using adaptive calculation of these internal variables than could be with pre-determined values of the same variables.

The external parameters can be described as an n-tuple, the values of all but 3 of which are held constant in this system. The interface (external) parameters are (P1) the approximate number of objects in the scene and (P2) the typical digitized complexity of the objects. An optional third parameter is (P3) the approximate fraction of background.

The interface parameter P2 is based on object scale within images. [Koenderink 84] discusses the limits of scale for images and objects. The outer scale of an image limits the maximum size of objects in camera coordinates, 512 pixels long in this case. The inner scale is the limit of camera resolution, typically 1 pixel. The inner scale determines the

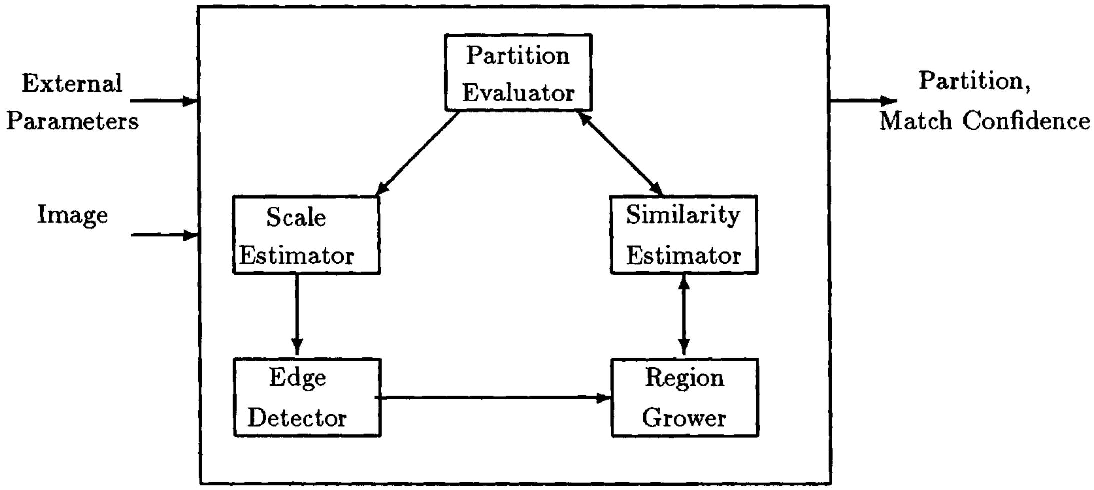


Figure 1: Model-Driven Segmentation Process

Model	Parameter	Instantiation	Range
Limit Number of Regions	Estimated Number of Regions	P1 parameter	[2,250]
Limit Complexity of Regions	Estimated Digitized Complexity (P:A)	P2 parameter	[0.008,1.0]
Limit Minimum External Scale	Fraction of Background	P3 parameter (or default)	[0.0, 0.999]
Topology of Regions	Neighborhood Connectivity	4-connected (constant)	
Borders are Edges	Algorithm Modules	(constant)	

Table 1: Models and External Variables

	P/A		P^2/A	
Domain	Min	Max	Min	Max
8 pt. Roman Nonie Text†	0.83	1.00	3‡	75
15 pt. Boldface Nonie Text	0.38	0.47	18	78

† Regions include letters and letter fragments (iff the whole letters are not 4-connected), but not holes in letters

‡ See [Rosenfeld 82] for a comparison of perimeter measurements which cause this value to be $< 4\pi$.

Table 2: Digitized Complexity Measures

spacing between two similar objects required for detection as separate regions. Objects in an scene also have an their own outer scale, which is the size of the objects. For a given object, the outer scale varies depending on where it is measured. The relationship of object outer scale to actual object size is set by the arrangement of the image acquisition system. Consideration is limited to object whose outer scale is between the inner and outer scale limits.

In this system, digitized complexity is defined as the ratio of perimeter pixels P to area pixels A, which is dimensionless. The perimeter is measured as the sum of the areas of the borders of the region, *i.e.* the pixels for which one or more of their 4-connected neighbors belong to other regions. Other measures of perimeter are described in [Ellis 79]. Perimeter pixels are included in region area. A more frequently used complexity parameter is p^2/A [Rosenfeld 82], where p has units of length. p^2/A is dimensionless and scale independent. For a given shape, digitized complexity is scale dependent. However, for a class of regions of similar width, such as all letters in a given font and type size, the digitized complexity P/A is similar, while p^2/A has more variation. In this system, an important consideration is the selection of appropriate edge scale, which is the outer scale of the interesting objects in units of the inner scale of the imaging system. Therefore, we have found P/A to be an effective external parameter for this system, specified as P2.

If the outer scale of objects in the scene varies over a wide range, then a third external parameter may be specified. This can be delineated as the approximate fraction of background (P3), or as the minimum acceptable region size (P4) as an alternative invocation. This parameter provides for adaptively applying outer scale non-uniformly across the image. It is used when the outer scale of the some regions (*e.g.* background) is much larger than the outer scale of other important regions in the scene. Since the background itself needs to be classified, it is not desirable to limit maximum size.

The specified external parameters are used to delineate global scene models. In principle, they could be applied locally (*e.g.* to smooth individual regions) after their global application. The alternative next step is to add semantic information to produce better results. The discussion here is based on the following assumptions:

- Stationary observer and non-moving scene
- Known scene scale
- Conditions above are constant during the time of observation

The tuple (P1,P2,P3) and the constant parameters shown in Table 1 act as constraints on the number of possible partitions of the pixels. In addition, the models of segmentation itself act as constraints, such as the model that the pixels in a region are connected. The topology of connection is also an external parameter, held constant in this work. The set of partitions which satisfies these constraints is still quite large. Several partitions of each image are considered; the final selection is made using a value function, which will be shown in Section 3.5. The value function itself is not a constraint, but simply a means of selecting the best partition.

3 Segmentation Modules

A tuple of internal variables is used to produce the partitions. Each internal variable (shown in Table 3) is associated with a particular module from Figure 1. The function of the segmentation system is to search the internal parameter space to find a partition which best matches the external models as measured by the value function.

The computationally intensive modules in the system are the Edge Detector and the Region Grower. Computation of the value function requires a complete segmentation, produced by the Region Grower, after edge detection is complete. Therefore, in this implementation, all of the internal values except local similarity (S, used by the Region Grower) are calculated, then a search is performed on S. Using this simplification, edge detection is done only once, and region aggregation is done approximately 5 to 20 times. However, all the other parameters are described, so that a more complex search through the parameter space could be performed. A planned development of the Model-Driven Segmentation Process is to vary edge detector scale and type.

3.1 Edge Detection Module

The edge detector produces thinned gradient edges by the method of [Canny 86]. Edge scale is estimated based on the ability of our camera system to deliver high quality edges, at the range of expected inner scale of objects. Though different scales will be appropriate for different image resolutions and camera systems, $\sigma_g = 2.8$ pixels is used here, because this provides good smoothing of perimeters while still delivering location accuracy. [Ellis 79] shows the level of improvement in the measurement of perimeter in noisy images using different levels of Gaussian smoothing. One problem with this scale is the tendency for edges to break at very sharp corners, which is shown in [Bergholm 87]. A planed development is the addition of multi-scale edge detection to the segmentation system, using larger scales. Smaller scale is already provided in the region aggregation module.

Since the gradient threshold G is set from scale parameters, gradient magnitudes are automatically normalized to provide maximum resolution within 8 bits.

Because the segmentation system is modular, any edge detection algorithm which yields single width edges could be substituted here without making other changes in the system.

3.2 Gradient Histogram Evaluation Module

Digitized complexity (P2) and fraction of background (P3) are used to estimate the expected total number of edge pixels in the scene.

This analysis applies the region/edge models as the following guiding heuristics

1. The strongest edge pixels are primarily object borders. For some types of scenes, this assumption does not hold, particularly if only one sensor is used. Further ex-

Table 3: Internal Variables for Different Modules

Internal Variables	Dependence	Instantiation Method	Module
local similarity S	external parameters, image noise/texture, image signal	search using feedback	Region Grower
minimum region size M	external parameters	calculated from external parameters	Region Grower
evaluation function	external model, internal model;	constant	Partition Evaluator
bias toward oversegmentation B_e,B_p	external model	constant	Edge Detector, Partition Evaluator
convolution mask shape	external model	constant	Edge Detector
convolution mask width σ_g	camera system; internal model	constant	Scale Estimator
gradient threshold G	external parameters	calculated from external parameters, gradient histogram	Edge Detector

perimentation with multiple sensors and/or multiple lightings may produce more reliable border edges. However, in the evaluation of the segmentation result, the match of the result to the model is checked.

2. The number of smoothed, thinned edge pixels is proportional to the number of border pixels an object, independent of the scale of the object. By the definition of border pixels as included in regions, there are 2 border pixels for each thinned edge, one for each region.

Use of this technique to threshold the gradient magnitude is analogous to a binary thresholding of a gray value image when the approximate fraction of pixels above threshold is known [Rosenfeld 82]. It was used by [Weszka 78] to minimize "busyness" of a binary thresholding. That technique is extended here to give the appropriate busyness of the complete segmentation.

In thresholding edges, this system's goal is to get as many border edges as possible and as few other edges. From a computational complexity criteria, it is less expensive to test for the removal of a border segment than to re-segment then test, based on the algorithms employed here. This is implemented as as the internal edge bias parameter, B_e. For this work, B_e is held constant at a value of 0.5. $|\{BP\}|$ is the number of border pixels, $|\{TP\}|$ is the total number of pixels, and $|\{EP\}|$ is the number of edge pixels.
$|\{BP\}| = |\{TP\}| * P3 * P2 * (1 + B_e) = 2 * |\{EP\}|$
T is set to satisfy $|\{EP\}|$ using a histogram of gradient values [Anderson 87].

3.3 Region Growing Module

The partitioning method used here is a form of edge-guided region aggregation. Regions are aggregated using a local similarity threshold S, as described in [Rosenfeld 82], except that aggregation is not permitted across or on edge pixels. Pixels in any region which does not meet the minimum size are not classified at first. To spread the initial regions, the S is then increased. Edge pixels and other unclassified pixels are aggregated with the initial regions. S is increased further until all pixels are classified. The initial region formation is not dependent on the choice of starting points, but only on the minimum region size and S. The spreading process is influenced by the rate at which the S is increased and by the choice of starting points. However, by increasing S slowly at first, the choice of starting points has very little effect, and the borders are accurate. A 3 step increase in S is used to aggregate all points. The initial similarity, S, is the value which is estimated and improved in the Similarity Estimation Module.

The details of border shape normally emerge during the spreading process, rather than during the edge-guided initial region formation. Thus the spreading allows borders to follow the inner scale details of the object. This occurs on high intensity difference edges which are more than 1 pixel wide. Initially, regions form on both sides of an edge, leaving an unaggregated strip. During spreading, the pixels are aggregated with the region closer in gray value. For example, if a corner is rounded as a result of the smoothing in edge detection, the pixels on the wrong side of the edge are left unaggregated. They then merge with the proper region during spreading. The advantage of using edges for outer scale is that the probability of detection of edge pixels on long borders is increased with smoothing, thereby preventing improper merging during the initial region formation. This occurs when there is a low contrast between two adjacent pixels across a border, particularly if the border is a gradual change in intensity. This particular error, known

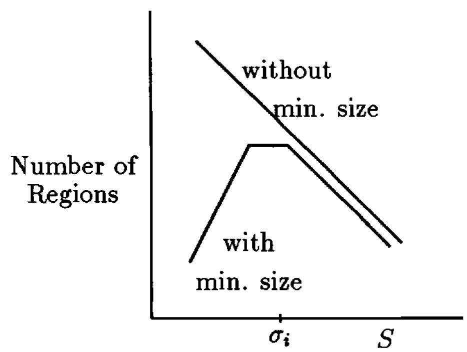


Figure 2: Region Formation versus Local Similarity S

as bleeding, is reduced by use of edges. Edges do not change the ability of the system to segment along step edges borders, although true step edges (1 pixel wide) cannot occur in any of our camera systems (except laser images).

The minimum region size, M, is based on P1 and P3. In this implementation, $M = \frac{1}{20} * \frac{P3*|\{TP\}|}{P1}$. Instead of using P3 and P1, the parameter P4 can specify M directly, and P3 is not used. Limiting the minimum region size is a frequent practice in segmentation [Haralick 85], but the exact minimum size threshold is difficult to justify on a theoretical basis. For example, [Beveridge 87] uses a minimum region size of 3 pixels, because smaller regions increase computational burden on later processing. A scale limitation on minimum region size seems to be a natural justification. However, since a choice of minimum region size must either be *ad hoc* or be determined by semantics, an external parameter with a back-up default value is used here, based on experience using scale.

3.4 Similarity Estimation Module

For an image which has detail at all scales and a consistent noise level (σ_i) throughout, the number of regions increases when S is reduced, and decreases when S is increased, except when $S < \sigma_i$.

Experience has shown that when $S < \sigma_i$, P1 may be met, but borders found are arbitrary and do not match edges. An example of this is shown in Section 4. However, for an unknown image, there is no estimate of the value of σ_i. To avoid an exhaustive search through S values, this heuristic is used to try values automatically, based on Figure 2 and segmentation experience [Anderson 87]. Since the our camera system normally has a noise level such that $\sigma_i \leq 4$ [McKendall 87], we begin with $S = 8$, then use feedback on the number of regions to improve S, unless the initial aggregation of pixels is very small. In that case, S is automatically reset to 20, and feedback is used on the number of regions. A minimum of 5 and a maximum of 15 values of S are tried. Naturally, for another camera system, another search procedure may be more appropriate. Improvements in this module's search procedure will reduce the computational complexity of the process. Additional work in this area is planned as the system is developed further.

The selection of the output partition is done by the Partition Evaluation Module.

3.5 Partition Evaluation Module

The evaluation of segmentations is a difficult and poorly specified problem because segmentation is not done as an end in itself, but as part of a larger process. Even if a "correct" segmentation were available for comparison, this would only give a binary evaluation. [Levine 85] developed a performance vector for each region in a segmentation. [Riseman 87] proposed a scalar relational measure, which is a function of different attributes of regions and lines. However, some distance function with no clear theoretical basis must be imposed to use these attributes.

Segmentations are evaluated using the global model that edges and borders should match, since the quality of the segmentation given by these algorithms depends on the match of the data to this model. If the global model is not appropriate to the data, a different model and resulting algorithm should be used for segmentation. [Moerdler 88] evaluates and integrates the results of various "shape from" algorithms upon texel patches based a measure of the match between "shape from" algorithm models and the actual data.

The external parameter P1 is used to guide the search through parameter space for the best segmentation within the model. Since this parameter normally is not known precisely, consideration is given to all segmentations of n regions for which
$(1 - B_p) * P1 \leq n \leq (1 + B_p) * P1$
For this work, B_p is held constant at 0.5.

Let EB be the set of edge-confirmed border pixels, and UB be the set of unconfirmed border pixels. Confirmation of a border pixel occurs when a thresholded, thinned gradient edge pixel occurs directly on or in the 4-connected set of a border pixel. An internal value, V, of a partition is set to:
$V = |\{EB\}| - B_p * |\{UB\}|$

After the final partitioning, the edge/border match is used as a measure of confidence in the borders, CM.
$CM = \frac{|\{EB\}|}{|\{EB\} \cup \{UB\}|}$
CM is used as a global confidence measure; however, it can be applied locally to obtain confidence in a particular region.

4 Results

4.1 Images

Real images from several different domains were tested, including a wide variety of signal level, noise level, texture. Since there is no standard for machine vision analogous to the IEEE standards for television, the process of setting internal thresholds automatically is made much more difficult. However, this work attempts to cover a variety of image types. All images are originally 512 by 512 pixels by 256 intensities. Because of the limitations of the edge detector and the camera system, the outermost 40 rows and columns are not segmented or shown. For display of the results, several images are reduced further as noted below, since the interesting material covers some subset of the remaining image area.

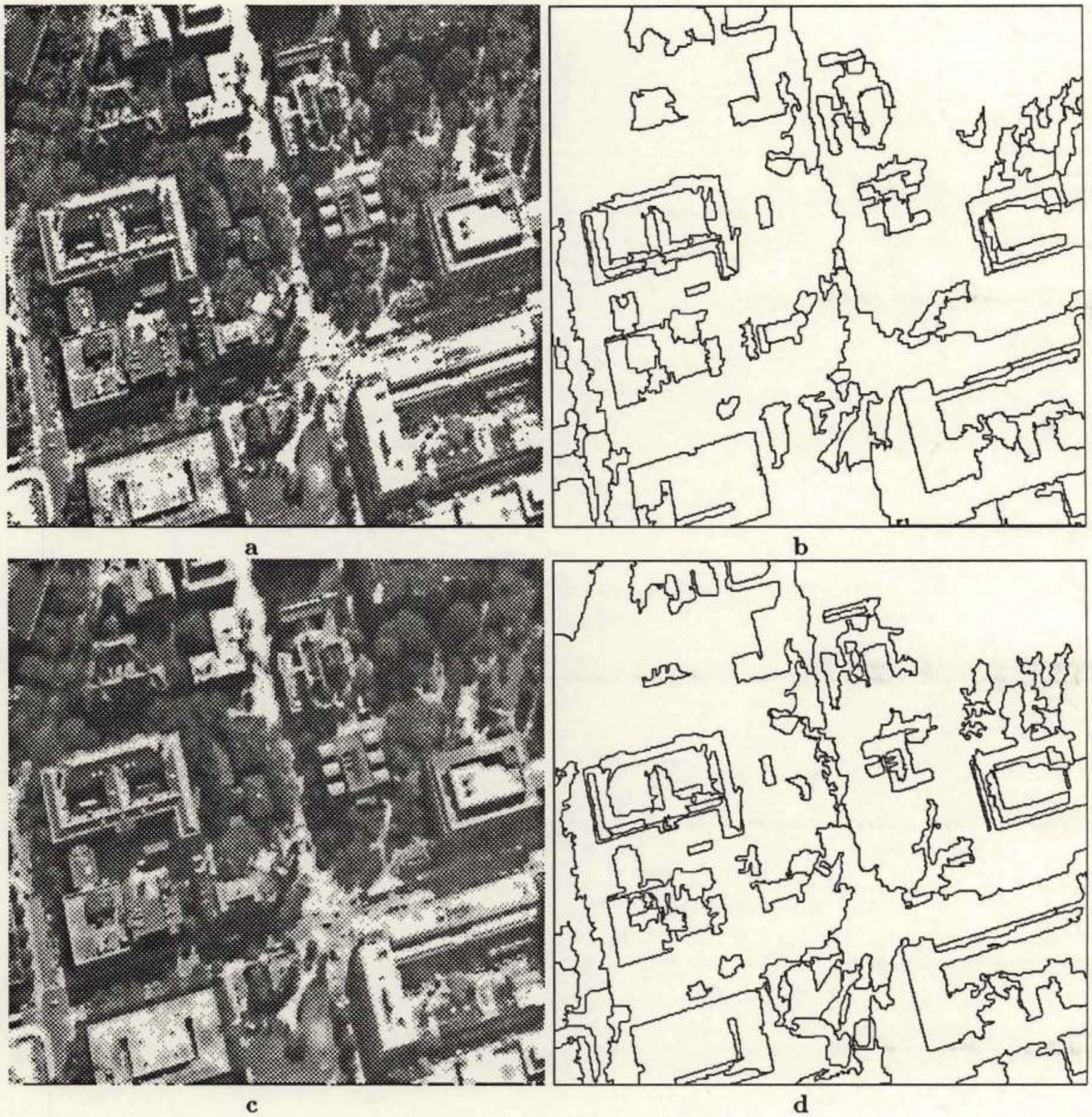
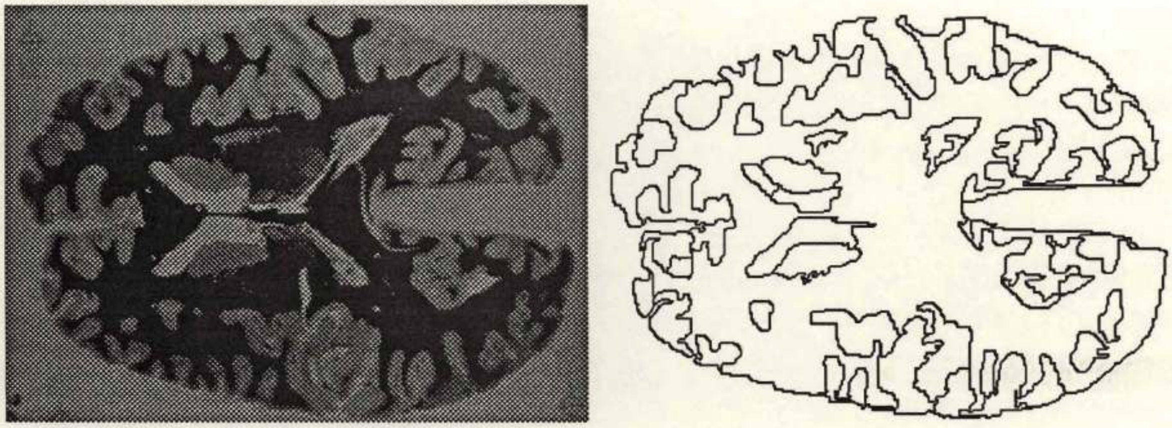


Figure 3: Aerial Photo
University of Pennsylvania Campus Original (**a**), Segmentation (**b**) Smoothed Original (**c**), Segmentation of Smoothed Image (**d**) Though the final segmentations are similar, the internal similarity parameter is automatically set to $S = 14$ for **a** to produce **b** and to $S = 5$ for the smoother **c** to produce **d**.



a b

Figure 4: Myelin Stained Brain Tissue Section

Original (**a**) and Segmentation (**b**)

Goal: 50 Regions. Found: 63 Regions CM = 77%

The background in the upper left corner contains a gradual intensity change of 50 (out of 256), which is appropriately not segmented. However, there is also a gradual intensity change in the top center from background into gray matter, which caused some gray matter borders to be missed.

The images include:

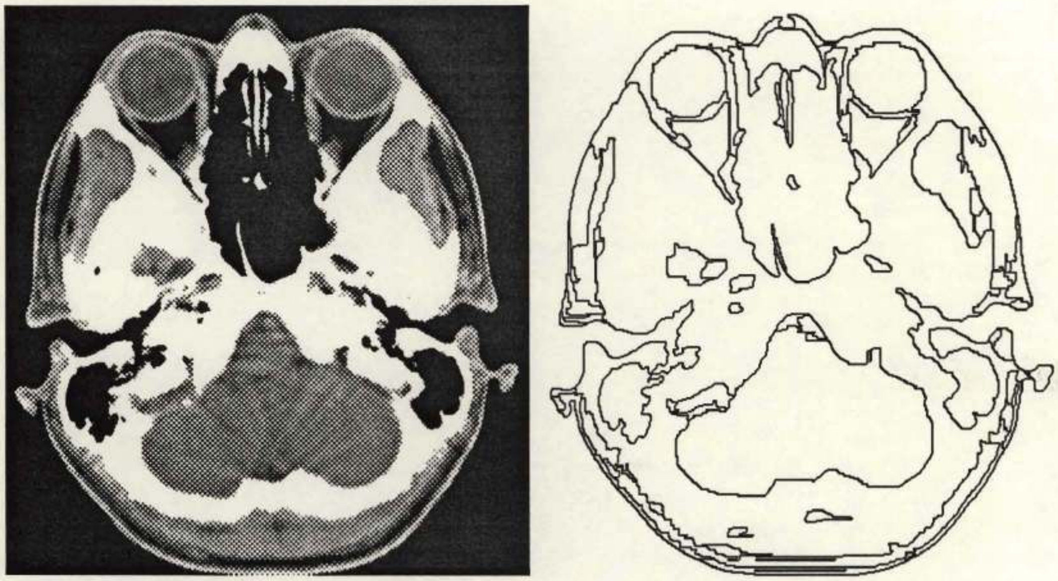
Aerial photographs of an urban area Figure **3a** contains signal at scales from below the limits of the image acquisition system to above the limits, and many scales in between.

Myelin stained human brain tissue section Figure **4a** contains many pixels of 0 intensity, which correspond to the myelin stained white matter. The image is unevenly illuminated, such that there is a gradient of 50 intensity values in the background of the area used, with the brightest values at the center of the image. Only 238 by 324 pixels are shown, with a ruler, tape and border of the paper removed, though the entire image is actually segmented.

Laser image of an architect's scale model of an urban scene This image comes from a single plane, structured light, laser range imaging system. It contains intensity values from 0 to 50, with almost no noise. Intensity values correspond to object height of 1.5 mm per intensity unit. The limited values reflect the limit of real object height difference, not a system limitation. Because the image appears very dark, it is not reproduced here. This image is the only one which contains true step edges.

CT scans of a brain Figures 5a and 6a are normalized to 256 intensity values such that there is maximum resolution for gray matter/white matter differences. Intensity values above 256 (*i.e.* bone) are mapped to 256. Some black background has been removed.

small objects on plain and textured backgrounds These include coins, toys, tools and kitchen objects. Backgrounds are poster board and solid and printed cloth of



a b

Figure 5: CT Scan of Human Head
Original (**a**) and Segmentation (**b**) Goal: 80 Regions, Result: 81 Regions with CM = 75%

various types and brightnesses. Figure 10**a** shows pennies on poster board. Figure 7**a** shows coins on Liberty cloth.

4.2 Predictions and Performance

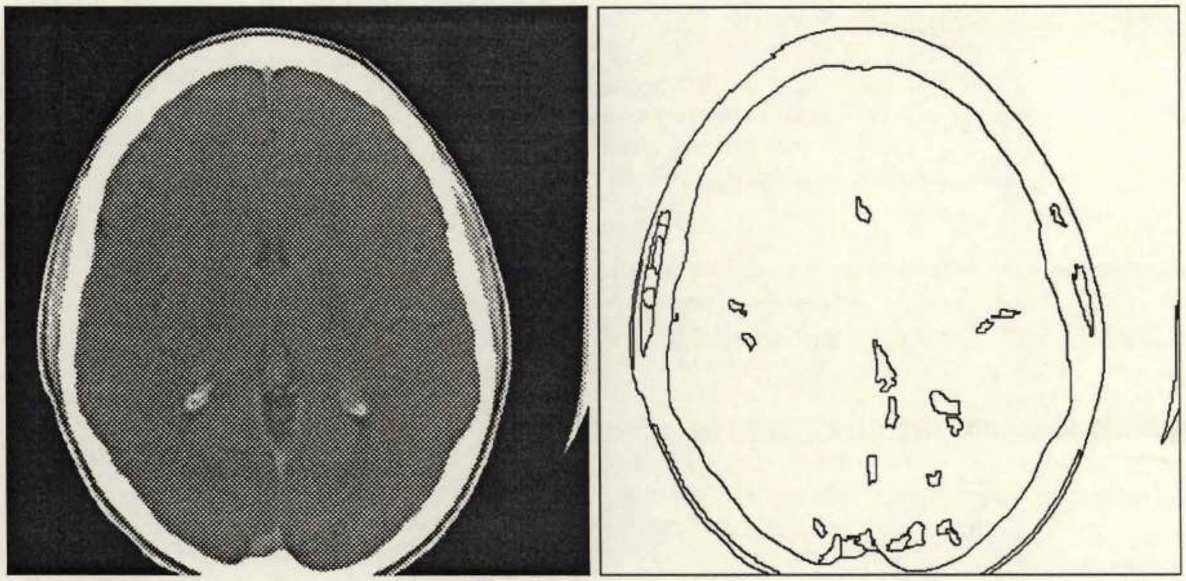
Several predictions can be made concerning the performance of the segmentation system:

1. Segmentation of regions with uniform brightness and with piecewise linear brightness and additive noise occurs iff the gradient of borders exceeds the gradient within regions.

   (a) When the background in an image contains closed areas of higher gradient than the objects desired, the background is segmented better than the desired objects. If the external parameters are adjusted to accept both types of regions, both types are segmented. An example of this is shown in Figure 7**b** and **c**.

   (b) Two adjacent regions of equal average brightness with different levels of texture will not be segmented. An example of this situation is the separation of gray matter from white matter in a CT scan, shown in Figure 6**b**.

   (c) Gradual changes in brightness will not be segmented. An example of this is shown at the top of Figure 4**b**, where the illumination changes across the image, brightest in the center.



a b

Figure 6: CT Scan of Brain
Original (a) and Segmentation (b)
Goal: 25 Regions. Found: 33 Regions CM = 79%
The confidence measure for brain/skull borders is high, but gray matter/white matter borders do not produce edges.

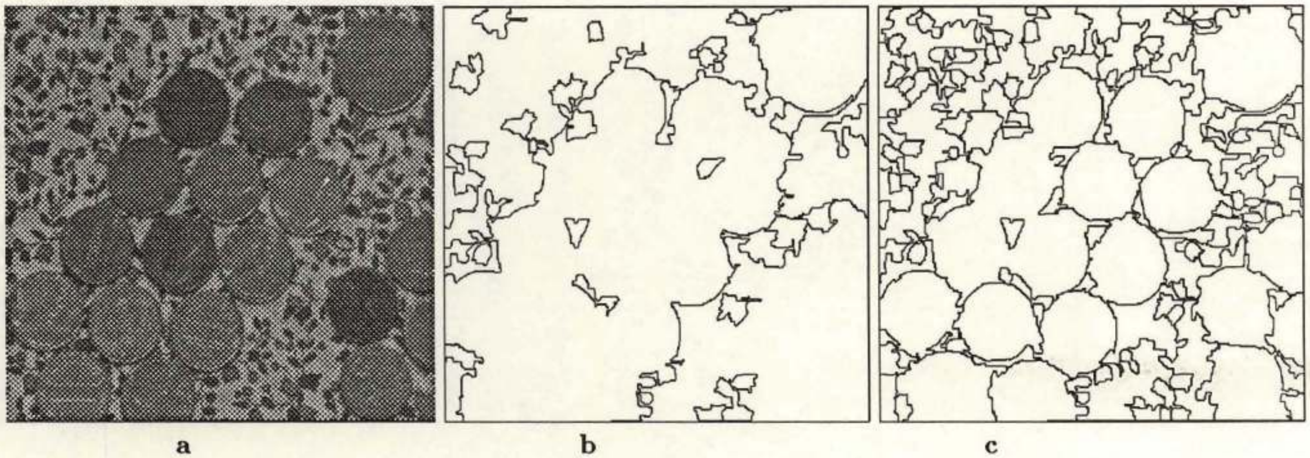


Figure 7: Coins on Liberty Cloth

Original (**a**). Segmentations:
b: Goal: 25 Regions, 50% background, Result: 34 Regions with CM = 75%.
c: Goal: 100 Regions with 20% background (default value), Result: 138 Regions with CM = 75%.

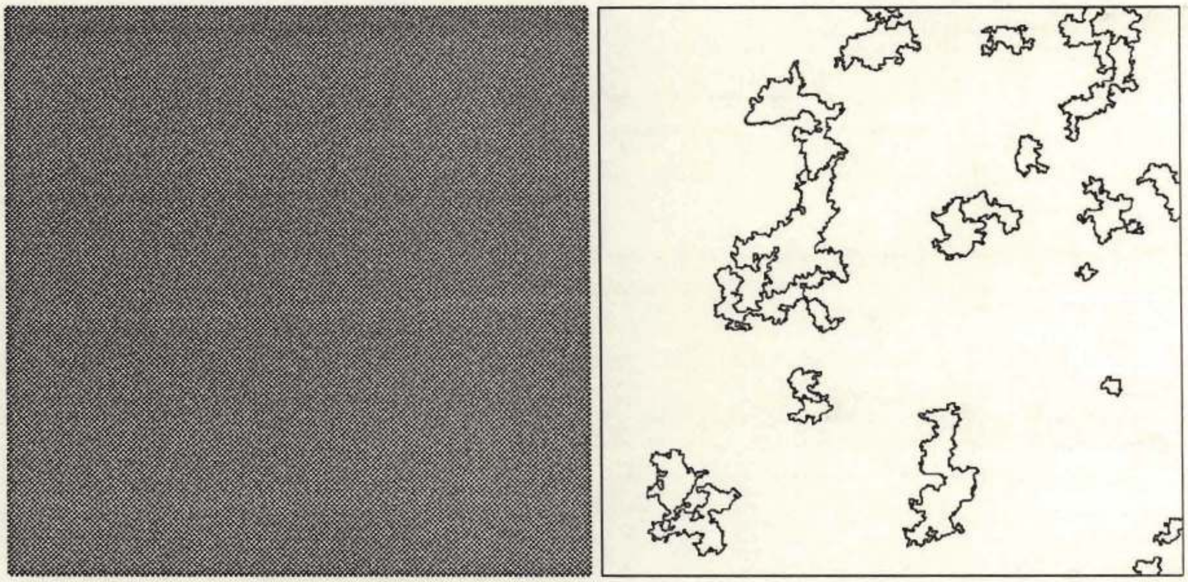
2. The ability of the system to divide two areas of the same brightness is limited by the separation between the areas. A minimum separation of 1 pixel is required by the Region Growing Module. The probability of detection increases as the separation increases, particularly when the separation is sufficient for the σ_g of the edge detector [Anderson 87]. The separation of objects from holes (*e.g.* areas of specular reflectance or closed interstitial spaces) depends on the size of the holes and the settings of the external parameters, since the Minimum Region Size is set from the external parameters.

3. The external parameters are robust, in that small changes in the parameters make small changes in the result. Table 3 shows the dependence of the modules on the external parameters.

4. *CM* confirms that the detected borders do indeed match the edges. Figure 8 shows the response of the segmentation system to a synthetic image containing only Gaussian white noise. Regions are detected, but *CM* shows that the algorithm is inappropriate to segment this image. Of course, the definition of *CM* uses only borders found, not borders which should have been found. For example, in Figure 6**b**, the number of regions desired is not met. *CM* is high because the skull/brain edges are sharp. However, gray matter/white matter borders are not detected.

4.3 Implementation

The Model-Driven Segmentation System, shown in Figure 1, has been written in C and installed on a Sun 3/260 running the X Window System, Version 11 (a trademark of



a b

Figure 8: Noise Only

Original (**a**) and Segmentation (**b**)

Goal: 25 regions. $CM = 3\%$, which shows that the segmentation process or external parameters are inappropriate to this data.

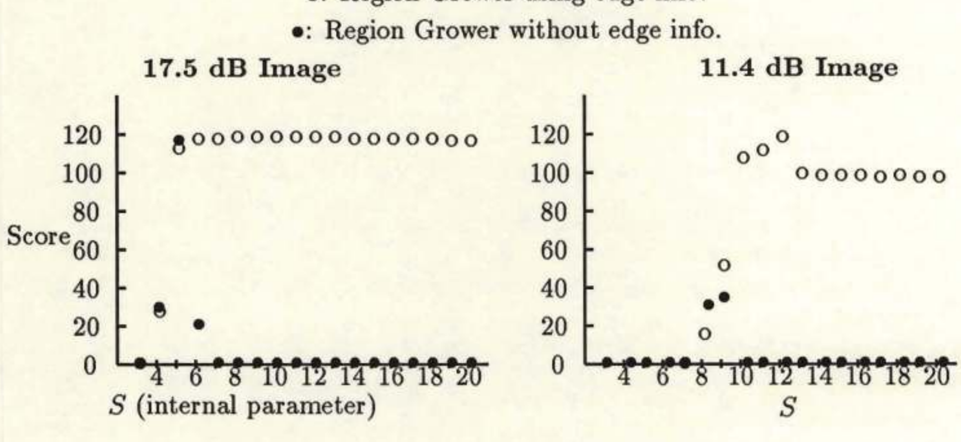


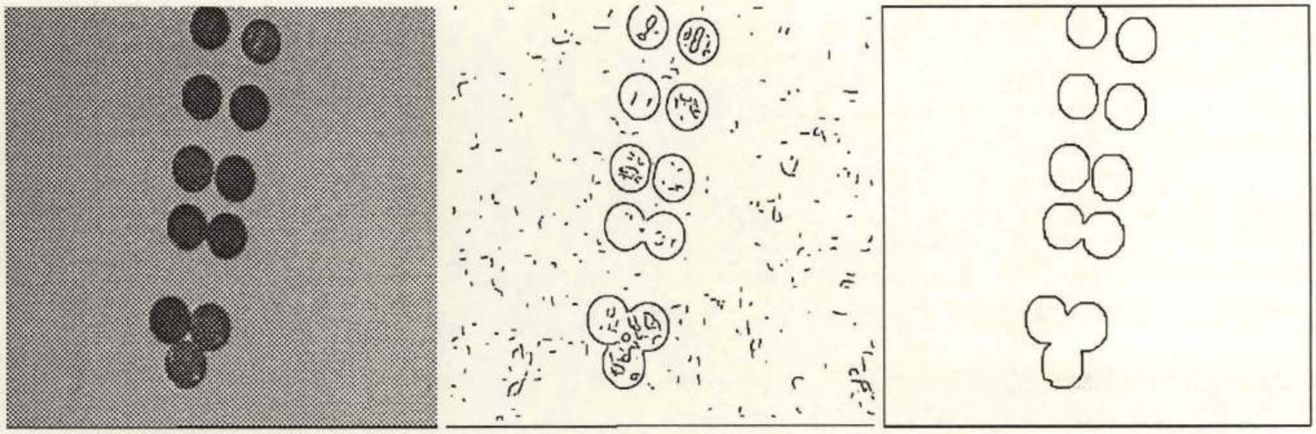
Figure 9: Region/Edge Combination versus Region Growing Alone
The score (described in Appendix A) is used to rate segmentations of the penny images with additive noise, using different local similarity (S) values, with and without assistance from edges.

---

the Massachusetts Institute of Technology) and displaying the original 512 by 512 by 8 bit image, thresholded edges, intermediate results and the final result on an HP Series 9000/320. The process, excluding edge detection but including displays, takes an average time of 10 minutes. It has been installed as a sequential process, but it could be partitioned within the region growing module and the region growing module could be run in parallel on several internal variables.

The segmentation quality from the region/edge combination algorithm implemented in the Region Growing Module is a significant improvement over either edge detection (and application of a connectivity operator to produce closed regions) or the basic region growing algorithm without edges. Using a scoring system and adjustments to the signal and noise which are described in Appendix A, Figure 9 shows the quality of segmentations of the top 6 pennies shown in Figure 10a, with additive noise and reduced signal as shown in Figure 11. For the 17.5 dB image, the 6 top pennies have unbroken edges; therefore, edges alone are sufficient to segment the top 6 pennies. Edges of the touching pennies are broken at all noise levels, and edges of some the top 6 pennies are broken in the 11.4 dB image. Therefore, the combination algorithm is better than edges alone. For the 11.4 dB example, the graph shows that best score of the combination is better than region growing alone.

The key improvement is this: the combination algorithm gives good results over a wide range of values of S. In other words, there is a small number of instantiations into the internal variable tuple which produce good results for region growing alone. The values of these variables depend on signal and noise, which are consistent throughout this particular image. This can be thought of as a small hypervolume in parameter space within which the segmentation score exceeds a threshold. For the combination algorithm, the hypervolume is larger, though still dependent on signal and noise. Complex images contain a variety of signal and noise levels. Each piece of border has a signal and noise value, and an associated hypervolume. A trial segmentation corresponds to a point in



a b c

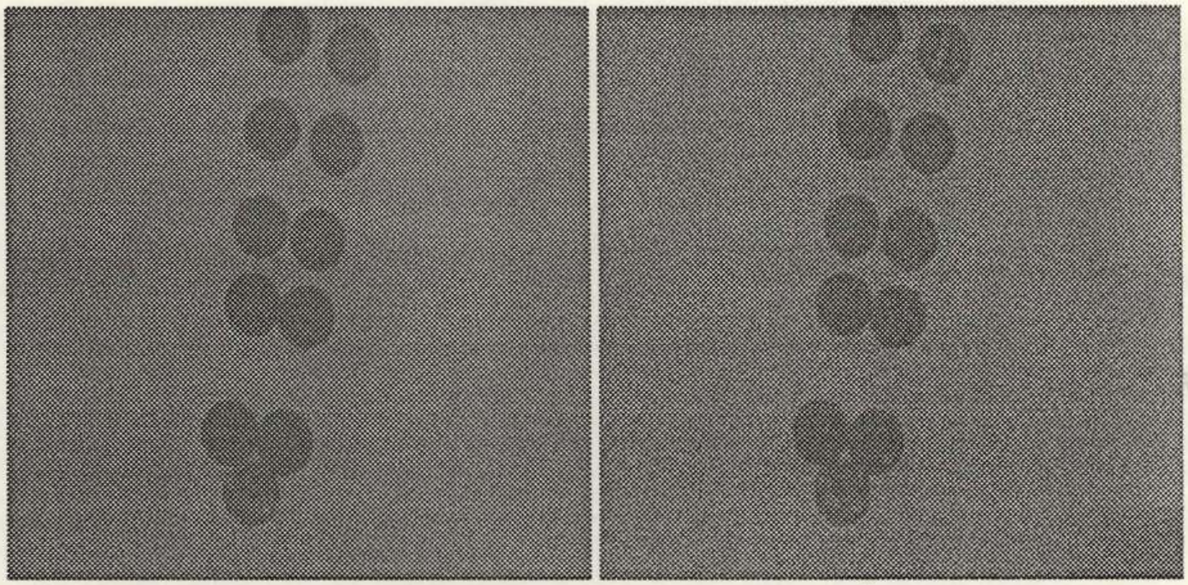
Figure 10: Pennies
Original (**a**), Thresholded Thinned Edges (**b**)
and Segmentation (**c**) of Pennies
Goal: 12 Regions, Result: 10 Regions. CM = 95%

parameter space, and its value is the number of hypervolumes containing the point. The combination algorithm, with larger hypervolumes, is likely to produce a better result.

Internal variables of S and T respond effectively to changes in image characteristics. This is shown in Table 4; an example of a change in S is shown in Figure 3.

Change	Gradient Threshold, T	Final S	Minimum Size	Confidence CM	Score	Test Image
Add Gaussian white noise	up	up	same	down	down	pennies
Reduce signal	same	same	same	down slightly	down	pennies
Smooth using Diffusion [Perona 87]	same	down	same	same	none given	Aerial Photo

Table 4: System Response to Changes in Input Image
For description of Score, see Appendix A



a b

Figure 11: Penny Images with Added Noise and Reduced Signal
a: Noise $\sigma = 5$, Signal ≈ 38, 17.5 dB
b: Noise $\sigma = 10$, Signal ≈ 38, 11.4 dB

5 The Limits of Bottom-Up Processing

We observed that the number of acceptable instantiations which satisfy a criteria decreases as the signal to noise ratio decreases. Acceptable instantiations are sets of values of internal variables from which segmentation results exceed a threshold upon a value function, such as the score in Appendix A. Therefore, the hypervolume (in internal parameter space) of acceptable segmentations decreases as the signal to noise ratio decreases. The search through parameter space may become more difficult, and it certainly becomes more important. This also explains the instability of segmentation programs when test images and parameter values are chosen in an *ad hoc* fashion. It should not be a surprise that internal variables and external parameters customized for one image do not work on other images, unless the signal, noise/texture and scale are similar.

Semantic information applied before segmentation is limited to scale parameters for several reasons.

- The 3 dimensional arrangement of objects is not predicted in advance because it varies from one scene to another.
- Illumination characteristics are not predicted in advance because they vary from one scene to another.
- Characteristics of camera noise are not predicted in advance because it varies over time for a single camera system, it varies from one camera system to another, and it may vary with intensity.

When the images are not well known in advance, semantic information applied before segmentation must be sharply limited. The scale parameters used here are: topology of regions (4-connected), integral geometry (estimate of the number of regions, within 50%), gross shape (perimeter to area ratio) and scale of analysis, as discussed in Section 2.1 and delineated in Table 1. Even though these parameters constrain the possible partitions, the result is still non-deterministic. Other variables are obtained from the image data. However, when more semantic information is available, it can be used to refine the segmentation and to characterize individual regions. We plan further work in this area.

The model that the gradient of borders exceeds the gradient within regions is a very general model which is met by many images. This system also gives good results on images with uniform intensity within regions and step edge borders, specifically laser images. However, since our other camera and lighting systems are not able to deliver uniform regions and step edges, we cannot depend on that model.

Any region intensity model for segmentation can be applied to any image. However, it is not likely to be appropriate to all images; therefore we confirm the appropriateness of our model automatically. We define and implement the confidence measure CM, which is the fraction of border pixels which are confirmed by the presence of significant edges.

From the design and theory of this modular process, we make predictions about the results, and we confirm them by experimentation. When the gradient of borders exceed the gradient within regions, the regions will be separated, providing they exceed the minimum region size derived from the scale parameters. The detection of two closely spaced objects of similar brightness is limited by their separation. If they touch, they well be partitioned as a single region. Two regions of different texture but the same average brightness will not be separated by this process. In addition, we show that the combination of edge and region information done in the Region Grower Module gives better results than either classical region growing or edge detection (with the application of a 4-connectivity operator) alone.

In this work, internal and external parameters are identified, specified and tested for robustness. An adaptive procedure for setting internal variables is presented and tested. The process is tested on images from several different domains. We believe that this explicit formulation of the segmentation process will provide a useful intermediate representation to higher level processes, such as integration of information from multiple processes (*e.g.* region grouping via texture and illumination, split and merge) and multiple sensors, and automatic object identification.

A Appendix: Evaluation of Known Segmentation

A human-guided evaluation of a known image is used to rank automatic segmentations using various parameters, additive noise and reduced signal. It uses an image of a group of pennies on a white background, shown in Figure 10**a**. Then the centroids of 6 of the pennies, confirmed by human guidance, are used as a known segmentation. Though statistical independence is not proved, the centroid does not depend on the same characteristics as the edge/border confirmation used within the system. Clearly, known segmentations are not available in most situations; however, this is used simply to understand the segmentation process.

The signal in the image is the intensity difference across the border of the penny, which averages 150 (out of 256). The actual transition zone of the change in brightness (width of the "step" edge) is 3 to 5 pixels wide. At the 2 closest pennies considered, the transition zones overlap, such that the signal strength is reduced approximately 25%. Several pennies have specular reflection which also reduce the effective signal by approximately 25%. An estimate of the signal is taken as the intensity difference from the center of each penny to a point approximately 2 penny radii from the center of the penny, then averaged among the pennies. The signal is reduced by requantizing the pixel intensities to a smaller number of values, then recentering the intensities within the original range. Spatially white, zero-mean, Gaussian noise is added after the signal is reduced. Approximately 40 combinations of signal and noise were tested, 2 of which are shown in Figure 11. To evaluate the result of the partition, the distance between the centroid of each detected penny and its standard centroid is measured. The standard centroid is produced by using the segmentation system on the original image. Each of the 6 pennies can obtain a maximum score of 20 (the approximate radius of the penny), and its score is reduced by 1 for each pixel distance from the standard centroid. Scores for the 2 images shown in Figure 11 are shown in Figure 9.

The pennies in the picture which are touching each other are not included in the 6 evaluation centroids.

References

[Anderson 87] H.L. Anderson, R. Bajcsy, M. Mintz. "A Modular Feedback System for Image Segmentation," University of Pennsylvania GRASP Laboratory Technical Report 110, June, 1987.

[Bajcsy 86] R. Bajcsy, M. Mintz, E. Liebman. "A Common Framework for Edge Detection and Region Growing," University of Pennsylvania GRASP Laboratory Technical Report 61, February, 1986.

[Bergholm 87] F. Bergholm. "Edge Focusing" in *IEEE Trans. on PAMI*, Volume PAMI-9, Number 6, November, 1987, pp. 726-741.

[Beveridge 87] J.R. Beveridge, J. Griffith, R. Kohler, A. Hanson, E. Riseman. "Segmenting Images using Localized Histograms and Region Merging," COINS Technical Report 87-88, University of Massachusetts at Amherst, October, 1987.

[Canny 86] J. Canny. "A Computational Approach to Edge Detection," in *IEEE Trans. on PAMI*, Volume PAMI-8, Number 6, November, 1986, pp. 679-698.

[Chou 87] P.B. Chou, C.M. Brown. "Multi-Modal Segmentation using Markov Random Fields," in *Proceedings of the DARPA Image Understanding Workshop* February, 1987, Volume II, pp.663-670.

[Cohen 87] F.S. Cohen, D.B. Cooper. "Simple Parallel Hierarchical and Relaxation Algorithms for Segmenting Noncausal Markovian Random Fields," in *IEEE Trans. on PAMI* Volume PAMI-9, No. 2, pp. 195-219, March, 1987.

[Ellis 79] T.J. Ellis, D. Proffitt, D. Rosen, W. Rutkowski. "Measurement of the Lengths of Digitized Curved Lines," in *CGIP*, Volume 10, pp. 333-347, 1979.

[Fua 87] P. Fua, A.J. Hanson, "Using Generic Geometric Models for Intelligent Shape Extraction," in *Proceedings of the IEEE Image Understanding Workshop* pp. 227-233, February, 1987.

[Haralick 85] R.M. Haralick, L.G. Shapiro. "Survey of Image Segmentation Techniques," in *CVGIP* Volume 29, pp. 100-132, 1985.

[Hartigan 75] John A. Hartigan. *Clustering Algorithms,* John Wiley and Sons, New York, 1975.

[Koenderink 84] J.J. Koenderink. "The Structure of Images" in *Biological Cybernetics* Volume 50, pp. 363-370, 1984.

[Levine 85] M.D. Levine, A.M. Nazif. "Dynamic Measurement of Computer Generated Image Segmentations," in *IEEE Transactions on Pattern Matching and Machine Intelligence* Volume PAMI-7, No. 2, pp. 155-164, 1985.

[McKendall 87] R. McKendall, University of Pennsylvania, personal communication.

[Moerdler 88] M.L. Moerdler, Columbia University, personal communication.

[Riseman 87] E. Riseman, A. Hanson, R. Belknap. "The Information Fusion Problem: Forming Token Aggregations Across Multiple Representations," COINS Technical Report 87-48, University of Massachusetts at Amherst, November, 1987.

[Rosenfeld 82] A. Rosenfeld, A.C. Kak. Digital Picture Processing, Academic Press, Second Edition, Volume 2, 1982.

[Pavlidis 88] T. Pavlidis, Y. Liow. "Integrating Region Growing and Edge Detection," Stony Brook Laboratory for Image Analysis and Graphics Technical Report 88-02, 1988.

[Perona 87] P. Perona, J. Malik, "Scale Space and Edge Detection using Anisotropic Diffusion," in *Proceedings of the IEEE Computer Society Workshop on Computer Vision,* pp. 16-22, November 30-December 2, 1987.

[Weszka 78] J.S. Weszka, A. Rosenfeld. "Threshold Evaluation Techniques" in *IEEE Transactions on Systems, Man and Cybernetics* SMC-8, pp. 622-629, 1978.

[Witkin 84] A.P. Witkin. "Scale space filtering: a new approach to multi-scale description", Chapter 3 in *Image Understanding 1984* S. Ullman and W. Richards, Ablex Publishing Co, 1984.