Edge Detector Evaluation Using Empirical ROC Curves

Kevin Bowyer
Department of Computer Science and Engineering
384 Fitzpatrick Hall
University of Notre Dame
Notre Dame, Indiana 46556
kwb@cse.nd.edu

Christine Kranenburg and Sean Dougherty
Department of Computer Science and Engineering
University of South Florida
Tampa, Florida 33620-5399
kranenbu@csee.usf.edu

Abstract

We demonstrate a method for evaluating edge detector performance based on receiver operating characteristic (ROC) curves. Edge detector output is matched against ground truth to count true positive and false positive edge pixels. A detector's parameter settings are trained to give a best ROC curve on one image and then tested on separate images. We compute aggregate ROC curves based on one set of fifty object images and another set of ten aerial images. We analyze the performance of eleven different edge detectors reported in the literature.

Keywords: performance evaluation, experimental computer vision, edge detection, comparison of algorithms.

Contents

1 Introduction

The need for sound experimental performance evaluation in computer vision is now widely recognized [7, 11, 39]. This paper focuses on evaluating the performance of edge detection algorithms. True positive and false positive edge pixels are counted based on comparison to manually-specified ground truth for real images. Performance is summarized using receiver operating characteristic (ROC) curves. We use adaptive sampling of edge detector parameter space to drive a train-and-test evaluation using a total of sixty real images.

The methodology and results presented are novel. No other work uses adaptive parameter sampling to construct empirical ROC curves for pixel-level performance evaluation, or has used so many real images, or has compared such a broad selection of detectors. The results indicate that relatively few of the more modern approaches offer any general performance advantage over the Canny detector. However, Heitger's "suppression and enhancement" detector does appear to offer a clear improvement. The results also indicate that edge detector performance rankings do not vary significantly between the types of imagery considered.

2 Related Work

A number of researchers have considered the problem of evaluating the performance of edge detectors. Table 1 summarizes some important elements of various related works. Related works can be categorized in broad ways that highlight elements of our approach that are unique relative to previous work and that we believe are important.

Some previous efforts can be categorized as "theoretical" or "analytical," in the sense that they are based solely on mathematical models of detectors and images [28]. The current state of analytical modeling of images, edges and detectors appears to be too primitive to allow confidence that this approach will produce results that are meaningful for real images.

A larger group of works is based on using synthetic images [1, 13, 14, 23, 27, 36, 38, 41]. Using synthetic images allows easy and precise specification of the ground truth edge locations. Evaluation can then be based on comparing detected edges to ground truth. However, the

Authors, **reference**	**Images Used** **In Evaluation**	**Comparison** **Metrics**	**Parameter** **Tuning**	**# of** **Detectors**
Deutsch & Fram [13] 1978	5 synthetic: ramp at 0, 15, 30, 45, 60 deg.	TP,FP-based metrics	subjective selection	three
Bryant & Bouldin [8] 1979	(a) 1 real (b) with 1 GT edge	(a) relative grade (b) absolute grade	(a) not practical (b) not described	five
Abdou & Pratt [1] 1979	theoretical model & 2 synth. images w. ramps	TP,FP-based metrics	none	seven
Kitchen & Rosenfeld [24] 1981	2 synthetic, but applicable to real	continuity & thinness metric	fixed sampling to optimize metric	seven
Eichel & Delp [14] 1981	one synth. (one real)	Pratt f.o.m. (subjective)	not described	three
Ramesh and Haralick [28] 1992	none – theoretical edge/noise model	ROC curve	heuristic sampling	two
Venkatesh & Kitchen [38] 1992	2 synthetic: 1 vert. and 1 diag. ramp edge	TP,FP counts and width, loc. metrics	fixed sampling to optimize metric	four
Spreeuwers & van der Heijden [35] 1992	1 synthetic, with Voronoi tesselation	"average risk," based on TP/FP metrics	fixed sampling to optimize metric	three
Strickland & Chang [36] 1993	1 synthetic, with vertical step edge	6-part qualitative and TP,FP-based metric	none	five
Jiang et al. [22] 1995	10/80 range images with manual GT	TP,FP-based measures	none	one
Kanungo et al. [23] 1995	1 synthetic, with vert. edge + noise	contrast threshold detection curve	not described	two *line* detectors
Cho et al. [10] 1995	1 real image, without GT	statistical metrics of confidence/likelihood	none	two
Zhu [41] 1996	1 synth. & 2 real, no pixel-level GT	metrics for width and connectivity	none	one
Palmer et al. [27] 1996	(a) 1 synth. with GT, (b) 5 real without GT	(a) line param. error (b) contrast metric	fixed sampling to optimize metric	one
Salotti et al. [30] 1996	2 real images, with manual GT	TP,FP-based metrics	fixed sampling to optimize metric	two
Heath et al. [19] 1997	28 real images, no pixel-level GT	human ratings of edge goodness	selection from 64 settings	five
Shin et al. [31] 1998	real image sequences	structure-from- motion results	adaptive sampling to min SFM error	five
Shin et al. [32] 1999	36 real images	ROC curve of 2-D object recog results	adaptive sampling for best ROC	five
Forbes & Draper [18] 2000	one synthetic	ROC curve of TP,FP pixels	adaptive sampling for best ROC	three
this work	60 real images, with manual GT	ROC curve of TP,FP pixels	adaptive sampling for best ROC	eleven

Table 1: Summary of some related work on methods for edge detector evaluation. Images used does not count variations such as adding different levels of noise, "# of detectors" does not count different implementations of the same detector.

synthetic images typically used contain only simple geometric patterns with added Gaussian noise. The essence of the complexity of a real image is that it typically contains edges of many different types, scales, and curvatures. Therefore, synthetic images are often far too simple to give confidence in the value of the results. Other researchers have expressed essentially the same opinion – *"... any conclusions based on these comparisons of synthetic images have little value. The reason is that there is no simple extrapolation of conclusions based on synthetic images to real images!"* [40]. Also, in our own previous related work we found that all of the edge detectors considered had essentially equivalent, and nearly perfect, performance on a simple synthetic image [19]. However, when those same detectors were compared using real images in a human rating experiment, significant differences were found.

Recently, Forbes and Draper have performed a study that uses the same basic ROC framework that we use, but with synthetic images generated from a graphics package [18]. Because this work relates so closely to ours, we will return to it in detail in the discussion section.

Another group of works uses metrics that reflect qualitative properties such as smoothness, continuity, thinness, and so forth [10, 24, 25, 27, 41]. Since these metrics do not require ground truth, they are readily applicable to real images. However, such qualitative metrics do not always appropriately reflect performance. For example, in the case of one smoothness metric [24], using a larger σ for the Canny detector leads to greater smoothness. But greater smoothing distorts edges near vertices and displaces edge location. Thus edge detector parameter settings that maximize such ground-truth-free metrics can result in poor edges. In a somewhat similar way, the "relative grading" metric [8] scores a detector according to how well it agrees with a suite of reference detectors. This avoids the need for specifying ground truth, but penalizes detectors that do not repeat the failings of the reference detectors! Metrics based on qualitative properties of detected edges may well be useful as additional secondary metrics, but they do not seem appropriate as primary performance metrics.

Only a few other works have used ground truth specified for real images. Bryant and Bouldin's [8] "absolute grading" metric used a single ground truth edge specified in an aerial image. One of our previous efforts looked at edge detection in range images and reported true positive (TP) and false positive (FP) statistics for comparison to ground truth on a set of 10 range images [22]. Salotti *et al.* used ground truth specified for two real images and reported (TP,FP) statistics [30]. This group of works is more closely related to our proposed method than the others mentioned above. However, there are two important differences between these approaches and our own approach.

One difference is that these approaches do not ensure that the parameters of each detector are equally well tuned. Various parameters in different detectors may have different ranges of allowed values, and different sensitivity in terms of changes in the detected edges. To account for this, the performance evaluation method must incorporate a step that samples the parameter space of each edge detector in an adaptive manner to find the parameter settings that represent the best performance for that detector.

A second difference is that simple (TP,FP) statistics alone may lead to an "apples and oranges" comparison. For example, a comparison of the Deriche and Sobel detectors found that the Deriche had slightly fewer missing edges than the Sobel, but at the cost of slightly more spurious edge pixels [30]. Since tuning a detector to increase the TP score generally also results in a higher FP score, this sort of result is difficult to evaluate. What is needed is to evaluate detectors over a range of the same TP values. This is the essence of the concept of the receiver operating characteristic (ROC) curve. Sound comparisons are more appropriately made using ROC curves rather than isolated (TP,FP) performance points.

In summary, the current state of the art in analytic modeling of images/detectors and the use of synthetic images seem inadequate. Similarly, by themselves, ground-truth-free "qualitative" properties of detected edges seem inadequate. The most useful method of evaluation would be based on ground truth for real images. Ours is the only approach to make such an evaluation at the pixel level by adaptively sampling edge detector parameter space to generate well-tuned ROC curves.

Shin *et al.* have explored task-oriented, rather than pixel-level, frameworks for edge detector evaluation [31, 32]. One approach evaluates detectors by the accuracy of the results when the edge map is used as input to a line-based structure-from-motion algorithm [31]. This work uses real images, but does not use pixel-level ground truth and does not involve ROC curves. Another approach evaluates edge detectors based on the results of a 2-D object recognition algorithm applied to a set of real images [32]. This work also does not use pixel-level ground truth. Such higher-level, task-based evaluations are certainly important, but are by nature more narrowly focused than our approach. In the future, we hope to explore the degree to which our pixel-level performance metrics can predict the results of task-level evaluations.

3 Experimental Materials

The raw experimental materials for this work are a set of edge detector implementations, a set of images, and manually-specified ground truth overlays for the images. This section describes each of these elements.

3.1 The Edge Detectors

The edge detectors considered here are those by Sobel, Canny [9], Bergholm [2], Sarkar and Boyer [33], Heitger [20], Rothwell *et al.* [29], Black *et al.* [4], Smith and Brady [34], Iverson and Zucker [21], Bezdek *et al.* [3], and Tabb and Ahuja [37]. The various detectors were selected as representative of approaches that are historically important and/or represent different interesting technical approaches. One important constraint was that we decided to evaluate a post-Canny detector only if an implementation was available that could be traced to the developer of the algorithm. We feel this is important in order to avoid questions of whether an implementation reasonably represents the algorithm developer's intent ([16, 15, 17]). Pointers to source code for most of the implementations can be found on our lab web page.[1] As used in our experiment, each detector read pgm format intensity images and output a single-pixel-wide binary edge map

[1] `http://marathon.csee.usf.edu/edge/edgecompare_main.html`.

in pgm format. Each detector has one or more parameters that were used to tune the sensitivity of the results. The ranges of parameter values sampled for each detector are listed in Table 2.

The Sobel detector is one of the earliest approaches to edge detection and is is included as a sort of historical reference point. The Sobel implementation was written at USF. Thresholding the Sobel edge strength image typically produces "thick" edges. Since we want to compare detectors based on single-pixel-wide binary edge maps, our Sobel implementation was extended to use the Canny non-maxima suppression and hysteresis routines. Thus the implementation has two parameters T_{hi} and T_{lo}.

The Canny implementation was re-written at USF based on an implementation that traces back to the University of Michigan. It uses standard non-maxima suppression and hysteresis routines. It has three parameters: σ to control the amount of smoothing, and T_{hi} and T_{lo} to control the hysteresis.

The Bergholm implementation was adapted from the Candela package available from KTH Stockholm. The Bergholm detector follows an "edge focusing" principle. First, "significant" edges are found with an edge strength threshold, T, and smoothing at a coarse scale, σ_{hi}. Then, the locations of these significant edges are tracked to a finer scale, σ_{lo}. Thus this detector has three parameters.

The Rothwell detector was translated into C at USF, based on C++ source obtained from the developer. This detector seeks to improve on the Canny by having better topology, in the sense of connectedness of edge chains. First, the image is smoothed using a Gaussian of size σ and gradient information is computed. Then all pixels with gradient strength greater than T, and which pass a non-maxima suppression step, are considered as initial edge pixels. Then all pixels with gradient strength greater than $\alpha \times N_i$ where N_i is the gradient strength of the nearest initial edge pixel, are added to the set of edge pixels. In this way, there is a gradient threshold that may vary over the image. Because the edges may now be thick, this is followed with an edge thinning algorithm. The detector has three parameters: σ, T and α.

	Technique	Parameter 1	Parameter 2	Parameter 3
Sobel 1970	non-max and hysteresis added	$T_{lo} = 0.0\text{-}1.0$	$T_{hi} = 0.0\text{-}1.0$	–
Canny [9], 1986	filter, non-max and hysteresis	$T_{lo} = 0.0\text{-}1.0$	$T_{hi} = 0.0\text{-}1.0$	$\sigma = 0.5\text{-}5.0$
Bergholm [2], 1987	scale-space focusing	$\sigma_{begin} = 0.5\text{-}5.0$	$\sigma_{end} = 0.5\text{-}5.0$	$T = 0.0\text{-}60.0$
Sarkar [33], 1991	optimal zero-crossing filter	$\beta = 0.5\text{-}5.0$	$T_{lo} = 0.0\text{-}1.0$	$T_{hi} = 0.0\text{-}1.0$
Heitger [20], 1995	suppression and enhancement	$\sigma = 0.5\text{-}5.0$	$T = 0.0\text{-}50.0$	–
Rothwell [29], 1995	Canny with topology added	$\sigma = 0.5\text{-}5.0$	$T = 0.0\text{-}60.0$	$\alpha = 0.0\text{-}1.0$
Iverson [21], 1995	logical - linear	$N_D = 4\text{-}24$	$T = 0.0,0.05$	–
Smith [34], 1997	univalue segment assimilating	$T = 1.0\text{-}50.0$	–	–
Ahuja [37], 1997	integrated edges and regions	σ_g lifetime	–	–
Black [4], 1998	robust anisotropic diffusion	$0.1 - 3.0 \times \sigma_e$	–	–
Bezdek [3], 1998	geometric Takagi-Sugeno 4	$\tau = 0.05\text{-}4.95$	$T = 0\text{-}255$	–

Table 2: Ranges of Parameter Values for the Detectors Analyzed.

The Sarkar-Boyer implementation was obtained from the developers. The detector is based on an optimal infinite impulse response (IIR) filter that responds to zero-crossings. One parameter, β, controls the scale of the filter in a manner analogous to the σ in the Canny detector, and a hysteresis step is applied, controlled by T_{lo} and T_{hi}.

The Heitger implementation was obtained from the developer. This detector uses even and odd symmetric filters in a “suppression and enhancement” approach. For each orientation of the filter, a response is obtained for an assumed step (line) edge. Then the first and second derivatives of the response in a direction orthogonal to the filter orientation are used to suppress responses that do not match that of the assumed edge type and enhance responses that do match. A number of parameters are available in the Heitger implementation. The results presented here are obtained adapting only the σ for the filter size and the threshold response magnitude T. Initial experiments showed that also adapting the parameter for the number of filter orientations gave only a small improvement in performance, and at the cost of greatly increased computation [12]. Other parameters were left at their default values.

The detector developed by Smith and Brady has the acronym SUSAN, for "Smallest Univalue Segment Assimilating Nucleus." It computes an edge strength measure based on the number of pixels within a local window whose intensity value is within T of that of the center pixel. It also computes an edge direction based on the center of mass of these pixels, and uses this to perform non-maxima suppression. Thus the only parameter of the SUSAN detector is the threshold, T. The detector uses either a 3x3 window or a larger approximately circular window. The authors recommend use of the larger window [34]. However, in our comparative evaluation of the two windows on a set of 10 object images and 10 aerial images, using the 3×3 window gave better results on 18 of the 20 images and approximately equivalent performance on the remaining two images. Thus all results presented here use the 3×3 window. The implementation obtained from the University of Oxford web page was modified to directly output the single-pixel-wide binary edge map.

The Iverson and Zucker detector uses a "logical / linear" approach that is similar in concept to Heitger's approach. Logical rules are applied in combining the response of linear filters in determining edges. The implementation of this detector obtained from the authors outputs an edge strength image in postscript format. We convert the format to pgm and apply non-maxima suppression to obtain single-pixel wide edges. One parameter is used to control the number of directions (N_D) considered, and another parameter to threshold the edge strength. The implementation can detect positive and negative lines as well as edges, but was set to detect only edges in our experiments.

The robust anisotropic diffusion detector of Black *et al.* was implemented at USF, and results on a sample image compared with those of the author's implementation. This detector uses a robust statistic, the median absolute deviation (MAD), to compute a parameter, σ_e, that controls the diffusion operation and the threshold for declaring edge pixels. Since the MAD is computed from the image, this might be considered to be a "parameterless" detector. However, the sensitivity of the detector can be adjusted by scaling the computed value of σ_e. Inflating σ_e decreases the number of edges found, and deflating σ_e increases the number of edges. In our experiments, the number of iterations of the anisotropic diffusion is kept fixed at 100. Initial experiments indicated that so long as the number of iterations is "large," the value of the MAD has much greater influence on performance. Because this detector produces thick edges, a Canny-style non-maxima suppression routine was added in order to get single-pixel-wide edges.

Results for the integrated edge/region detector of Tabb and Ahuja were obtained by transferring image files to the University of Illinois to be processed there. The edge detection result files were transferred back to USF. One of the more interesting aspects of this detector is that it finds edges as the closed boundaries of regions of "similar" pixel values. This property is not true of many edge detectors, and may be important for some applications. This detector looks at each pixel in terms of the attraction force from it to other similar pixels within some spatial region. At the boundary of a region, adjacent pixels have force vectors that point in approximately opposite directions. The size of the spatial region is adapted at each pixel based on the magnitude and stability of the force vector. The implementation produces result files for a fixed set of nine values of one parameter (the "σ_g lifetime," see [37]).

The Bezdek *et al.* detector approaches edge detection as a sequence of four operations: image conditioning, feature extraction, blending, and scaling. It emphasizes understanding the geometry of the feature extraction and blending functions, and uses the Sobel kernels for features. The implementation obtained from the authors produces an edge strength image, and so non-maxima suppression was added to this implementation to produce a single-pixel-wide binary edge map. Two parameters were used with this detector: τ controls the blending function steepness at the origin, and T is a threshold on edge strength.

3.2 The Images

We used a total of sixty images in our experiments. Of these, fifty represent the general domain of generic object recognition from grayscale images, and ten represent the domain of aerial image analysis. To investigate whether detector rankings vary based on type of imagery, we report results separately for the two sets.

Each of the fifty images representing the domain of generic object recognition contains a single object approximately centered in the image, appearing essentially unoccluded, and set against a natural background for the object. The set of images contains both indoor (39) and outdoor (11) scenes, and both natural (8) and man-made (42) objects. These images were originally acquired as color images using a 35-mm camera and placed onto Photo CD by a commercial lab. Gray scale versions of the images were obtained from the three color planes using the formula: $intensity = 0.299 \times red + 0.587 \times green + 0.114 \times blue$. The images were then individually cropped to approximately 512×512 in size, and to have the object of interest approximately centered in the image.

The ten aerial images were sampled from the DARPA-IU Fort Hood aerial image data set. Some contain essentially vertical views and some contain more oblique views. All of the aerial images are 8-bits per pixel, and are approximately 512x512. These were cropped from larger original images in the DARPA Fort Hood aerial image data set.

Due to space considerations, the complete set of images is not presented in figures in this paper. However, all of the images and the ground truth templates used in this work may be obtained by down-loading the tar file available from our lab web page. The versions of the images viewable on the web page are in jpeg format, but the tar file contains the pgm format versions.

3.3 Ground Truth for the Images

Ground truth (GT) was manually created for each image. The GT overlay for an image is another image of the same size, in which black represents edge, gray represents no-edge and white represents “don’t care.” The GT is created by specifying edges that should be detected and regions in which no edges should be detected. Areas not specified either as edge or as no-edge default to don’t-care regions. This makes it practical to specify GT for images that contain regions in which there are edges but their specification would be tedious and error-prone (for example, in a grassy area). However, our method for constructing ROC curves does require that each image have a substantial amount of both edge pixels and no-edge region. Figure 1 shows, for each domain, an example image and its GT.

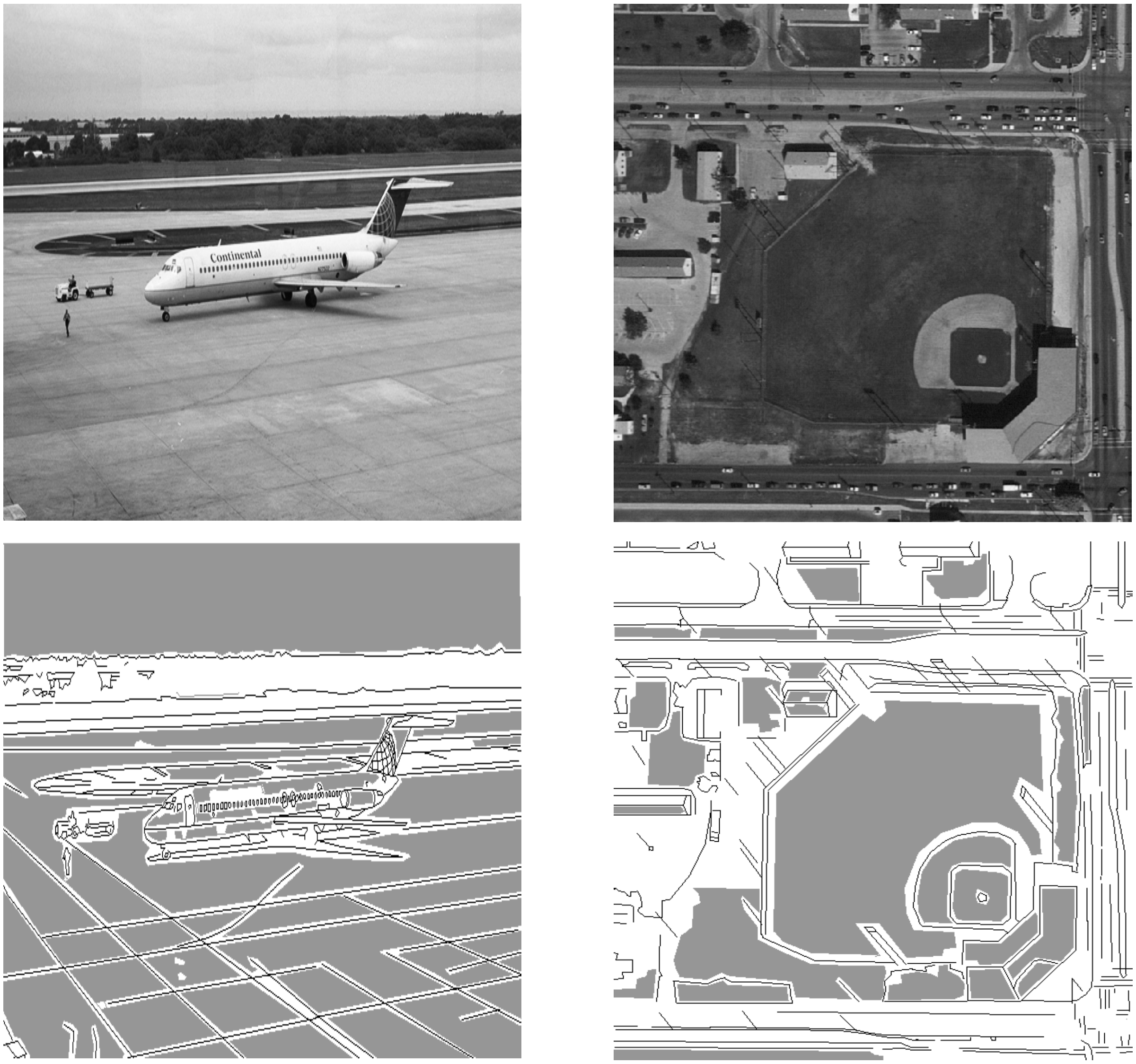


Figure 1: Example “Object” and “Aerial” Images and Their Ground Truths.

If a detector reports an edge pixel within a specified tolerance of an edge in the GT, then it is counted as a true positive (TP), or a “Matched GT Edge Pixel.” If a detector reports an edge pixel in a GT no-edge region, then it is counted as a false positive (FP). Edge pixels reported in a don’t-care region do not count as TPs or FPs. The ROC curve for a given image and edge detector is created from the (TP,FP) points representing different parameter settings of the detector.

If two different people specify GT for an image, or the same person specifies GT on separate occasions, the results will generally differ in their details. If this level of variation in the details of the GT caused the performance ranking of edge detectors to vary, then this approach to performance evaluation would have little value. Figure 2 presents the results of a check on this question. Five different GTs are shown for the same image, along with the ROC curve obtained using each GT. The ROC curves were created by the method to be described later. As plotted, a curve that comes closer to the lower left corner is better. The first two GTs were created by person A on separate occasions, taking approximately 30 minutes for the first and approximately 60 minutes for the second. The next two GTs were created by person B on separate occasions, again taking approximately 30 minutes and 60 minutes, respectively. The last GT was created by person C taking approximately 90 minutes.

It is readily apparent that the details of the different GTs vary. However, each GT has a substantial amount of both edge and no-edge. The first GT has approximately 13,000 edge pixels and 41,000 no-edge pixels, the second has 18,000 edge pixels and 66,000 no-edge pixels, the third has 8,000 edge pixels and 76,000 no-edge pixels, the fourth has 13,000 edge pixels and 65,000 no-edge pixels, and the fifth has 10,000 edge pixels and 112,000 no-edge pixels. The important point to note is that the relative ranking of the different detectors is essentially unchanged across the different GTs. The absolute position of the ROC curves on the plots varies, but the relative ordering of the results is stable. For each GT, the Heitger shows the best result, followed by the Canny and Bergholm, then the Smith, and then the Sobel. This indicates that the level of variation in performance ranking due to varying details in the GT is small comparison to the performance differences in between the detectors. In essence, once the set of training data for creating the ROC curves reaches 8,000 or more edge pixels and 40,000 or more no-edge pixels, minor changes to the GT seem to have no substantial effect on the relative ranking of the detectors.

4 Methods

There are three important elements of the experimental method. One is the comparison of edge detector output for an image to the corresponding GT to obtain a (TP,FP) count. The second is the algorithm for adaptively sampling the parameter space of an edge detector to obtain the training ROC curve for a given image. The third is the method for aggregating a set of ROC curves.

4.1 Comparing Detected Edges To Ground Truth

A core piece of software for the experiments is the "comparison tool." The inputs are an edge map and a GT image, and the output is a (TP,FP) count. The algorithm used is as follows. Each edge pixel in the detected edge map is evaluated in comparison to the GT. If the edge pixel falls in a no-edge region, then the FP count is incremented by one. If the edge pixel lies within T_{match} pixels of a GT edge, then the TP count is incremented by one, and the matched pixel in the GT is marked so that it cannot be used in another match. The T_{match} threshold for tolerance in matching a detected edge pixel to GT allows detected edges to match the GT even if displaced by a small distance. This is visible in the GT images as a small band of white around each black edge. The experiments reported here use a value of $T_{match} = 3$.

In general, a detected edge pixel may have multiple potential matches to a GT edge pixel. We take the closest-distance match within the T_{match} area. Draper and Forbes used the farthest-distance match in their implementation of our evaluation method [18]. Liu and Haralick looked at this particular matching problem from a theoretical perspective [26]. In our experience, these variations in the matching algorithm do change the the absolute TP and FP counts slightly, but do not affect the performance ranking of the detectors.

In order for the (TP,FP) count to be consistent and unambiguous, the GT should not contain specified edge pixels that are within T_{match} pixels of the border of a no-edge region. This condition is easy to enforce. If a detected edge pixel falls in a don't-care region and is not matched to a GT edge, then it does not contribute to either the TP or the FP count.

A problem arises in comparing detectors when they report results differently around the border of the image. For example, one detector may report results at all pixel locations in the original image, whereas another detector may report no results for a several-pixel-wide border around the edge of the image. The GT overlays were created to be the same size as the original image. If a detector leaves a border around the edge of the image where it reports no results, then it naturally cannot match any GT edges specified in that region. Also, it could not generate any FPs in portions of no-edge region along the border. Thus the (TP,FP) counts for the two detectors would not be directly comparable. Fortunately, the effect of this problem is generally small. A one-pixel border around a 512×512 image represents less than 0.8% of the total image area. Of the detectors considered here, the only one for which this could present a substantial problem is the Ahuja, which reports no result for a ten-pixel border of the original image. For this detector, the compare tool was used to generate (TP,FP) counts only over the area of the image where the detector reports results. Thus its results are actually computed over a smaller image area than the results for the other detectors.

4.2 The Training ROC Curve for an Image

For a given detector and a given image, a training ROC curve is found by adaptively sampling the detector's parameter space. The adaptive sampling terminates when it has found parameter settings that represent the family of best TP/FP tradeoffs. The adaptive sampling is done as follows. Initially, the specified range of each parameter is sampled by four uniformly-distributed values. If the detector has P parameters, this results in 4^P points in the parameter space. Each edge map is compared to the GT to obtain a (TP,FP) count for the corresponding parameter point. The resulting set of (TP,FP) points is plotted on a graph with "% Unmatched GT Edges" on the X axis and "% FP" on the Y axis (see Figure 3). In this format of the ROC curve, the ideal point is (TP,FP) = (0,0) and an ROC curve that lies to the lower left of another curve is better.

While each parameter setting generates a point in the ROC space, not all points necessarily lie on the ROC curve. A point appears on the ROC curve only if no other point has both a smaller % Unmatched GT and a smaller % FP. The upper left panel of Figure 3 shows a plot of 112 parameter points of the Canny edge detector applied to the airplane image of Figure 1. In this case, only a small percent of the sampled points actually lie on the ROC curve. This is because the Canny detector has three parameters and some combinations of parameter values result in poor performance tradeoffs. For detectors that have only one parameter, typically all of the sampled parameter values do generate (TP,FP) points that lie on the ROC curve.

Next, possible refinement in the sampling of each parameter is considered. Each parameter is individually sampled at the mid-points between current sample points. For example, if the detector has 3 parameters, then the initial sampling is $4 \times 4 \times 4$ and the first set of refinements considered is $7 \times 4 \times 4$, $4 \times 7 \times 4$, and $4 \times 4 \times 7$. The refinement that results in the greatest improvement in the ROC curve is kept. This adaptive refinement of the parameter sampling continues for at least two iterations, and until the improvement in the ROC curve falls below 5% of the area under the curve (AUC). At this point, the ROC curve is a good approximation to the ideal ROC curve. Figure 3 shows the ROC curve at stages of the refinement in parameter sampling.

The empirical ROC curve as we derive it generally has endpoints that fall short of the upper left (Unmatched GT = 0, FP = 1) and the lower right (Unmatched GT = 1, FP=0) corners of the plot. For most detectors and images, the TP level does not ever reach 1. Also, for most detectors and images, the FP level falls to zero in the range of Unmatched GT = 0.8 to 0.6. The upper-leftmost point on the ROC curve generally represents an extreme parameter setting for the detector; for example, the lowest edge strength threshold value considered. Also, it is generally found on the initial coarse sampling of the parameter space. The area under a given ROC curve is calculated over whatever range of TP values the sample points span.

A training ROC curve was computed in this way for each of the sixty images, for each of the detectors other than the Ahuja. For the Ahuja detector, we received edge image results representing a sampling of nine pre-determined values of the single parameter in the detector. The edge images were received in a format twice the size of the original image, with edges marked "between" original pixels. The over-size edge images were converted to the original image size at USF, and ROC curves constructed.

4.3 Test ROC Curves

We can consider the construction of ROC curves in a *train and test* paradigm. The ROC curve found by adaptive search of a detector's parameter space with a given image is a training ROC curve based on that image. The family of parameter settings that lie on the training ROC curve for a given image can be used to apply that detector to a different image to obtain a test ROC curve. For a detector with more than one parameter, the training ROC curve for a given image should be better than any of the ROC curves obtained when that image is used as a test for other images' training curves. For a detector with a single parameter, the difference between its training and test curves would come only from sampling the one parameter at a different granularity.

The use of a train and test paradigm is important when comparing detectors that have different numbers of parameters. It is possible for detectors with more than one parameter to rank lower in test results than in training results. However, our experience with the detectors evaluated to date is that the difference between a given detector's training and test results is generally small relative to the differences between detectors.

4.4 Aggregate ROC Curves

For a data set of N images, we have N different training ROC curves. For each training ROC curve, we can have $N-1$ different test ROC curves. Thus a fifty-image dataset produces $50 \times 49 = 2450$ test ROC curves, and a ten-image dataset produces $10 \times 9 = 90$ test ROC curves.

The total number of test ROC curves is too great to consider each individually for subjective visual evaluation. However, we can create an aggregate curve from a set of curves. The aggregate ROC curve is created by sampling each of the individual ROC curves at the same fixed set of TP values. In general, this requires linear interpolation between two empirically-obtained ROC points on a curve. The aggregate FP value is then simply the average of the individual FP values. The aggregate test ROC curve visually represents the average performance that can be expected from a detector by tuning it on one image and then using the tuned detector on a new set of images.

One complication to be aware of in interpreting the aggregate ROC curve is that not all individual ROCs contribute to the aggregate ROC across the entire range of TP values. Some individual GTs are "hard" for some detectors, in that the detector may not be able to obtain 100% of the TPs even at a very high level of FPs. Thus all the points on an aggregate ROC curve may not represent an average over the same number of individual curves. This can lead to a "spike" on the aggregate curve where the number of individual curves being averaged changes. This is evident, for example, on the aggregate curves for the Sobel detector in Figure 5.

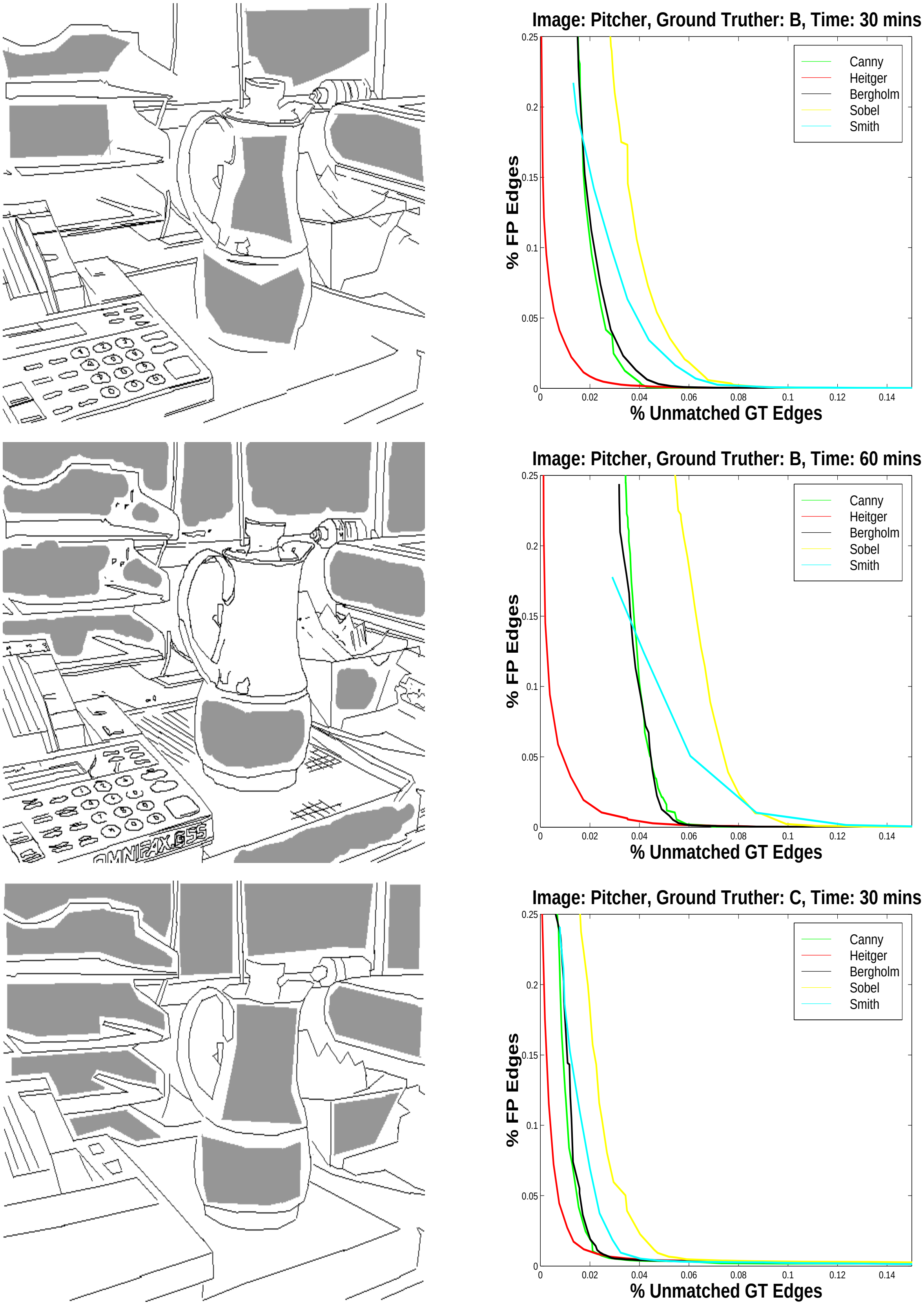


Figure 2: Performance Rank and Variation in the Ground Truth Specification.

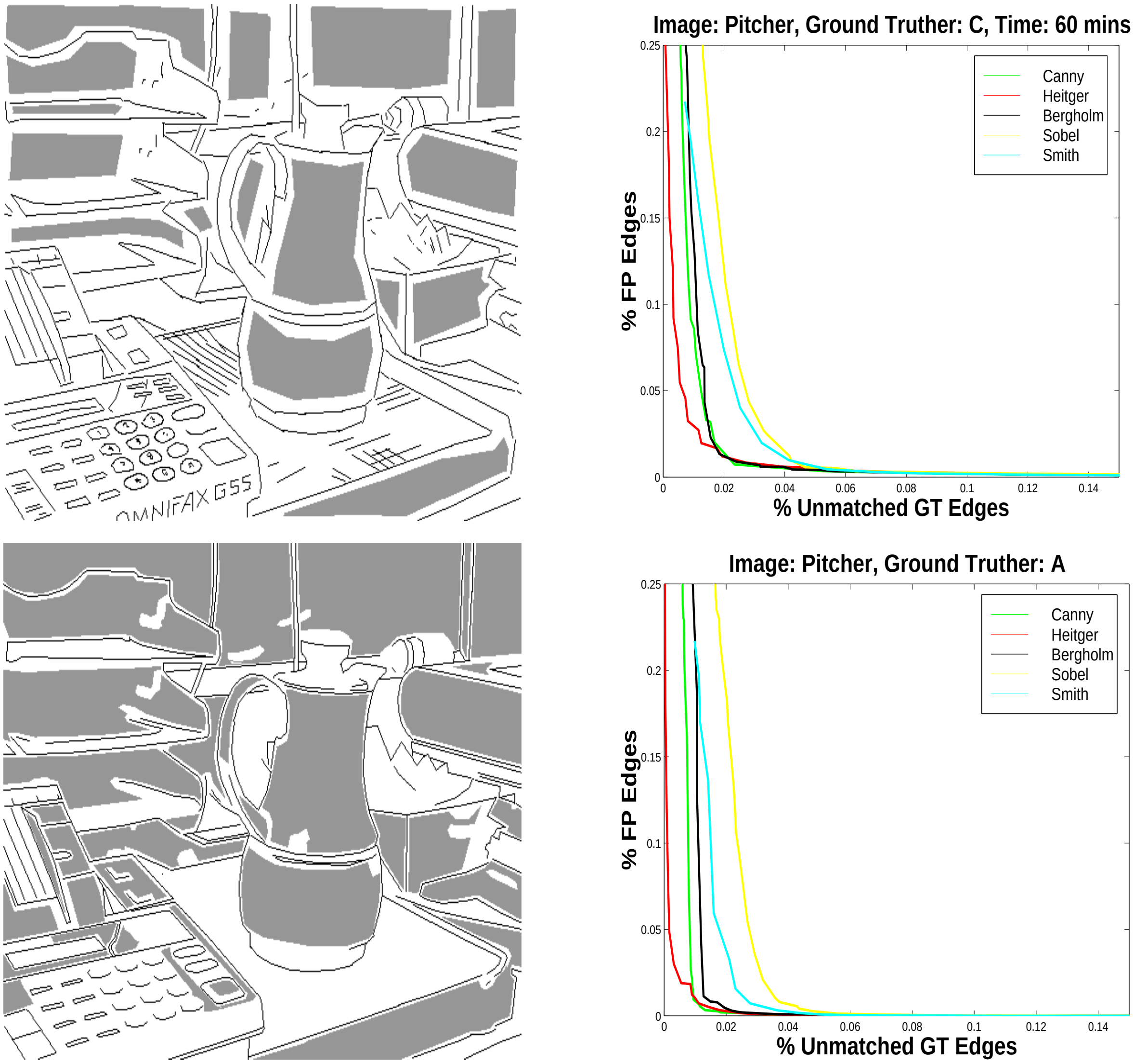


Figure 2: Performance Rank and Variation in the Ground Truth Specification.

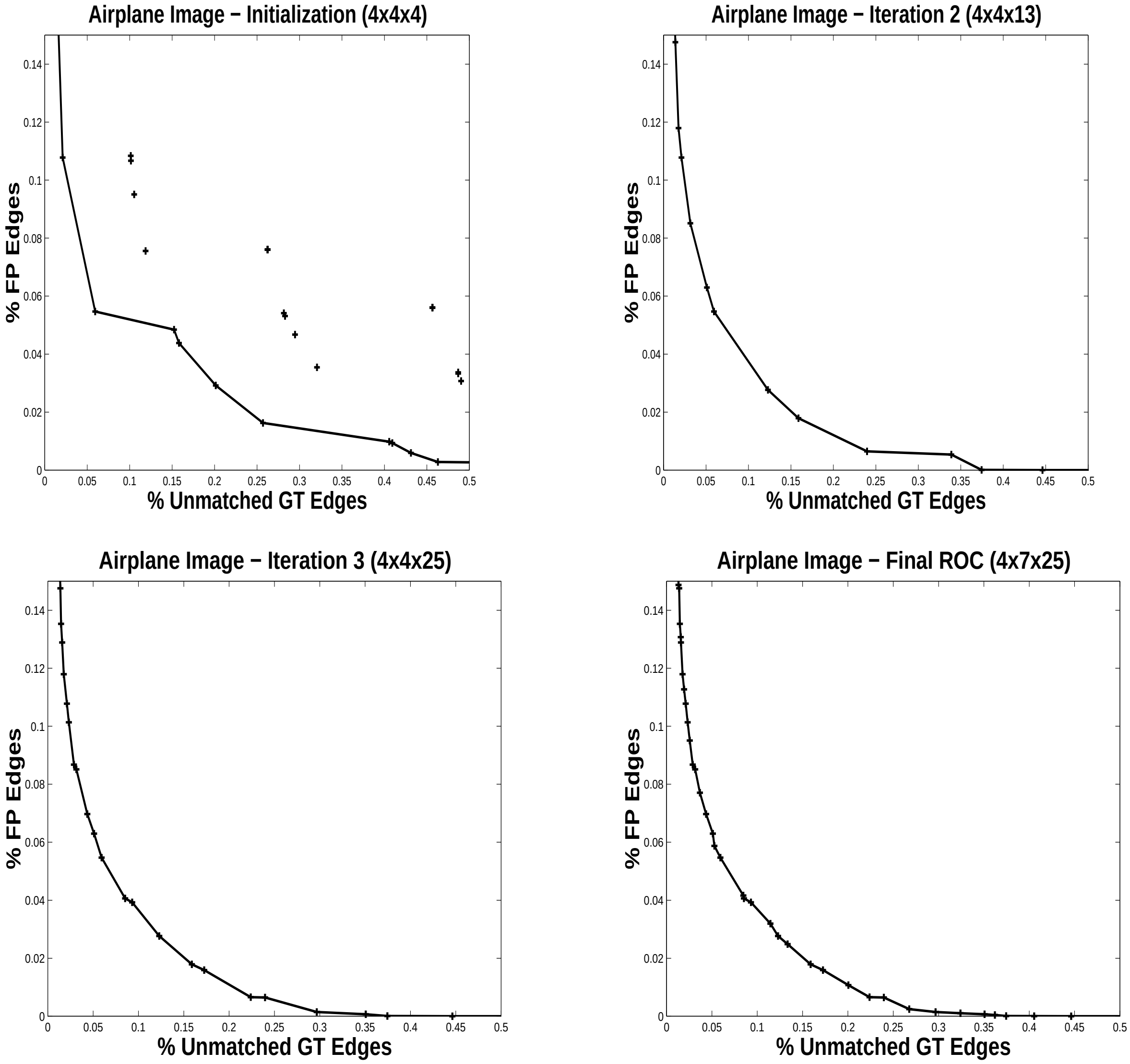
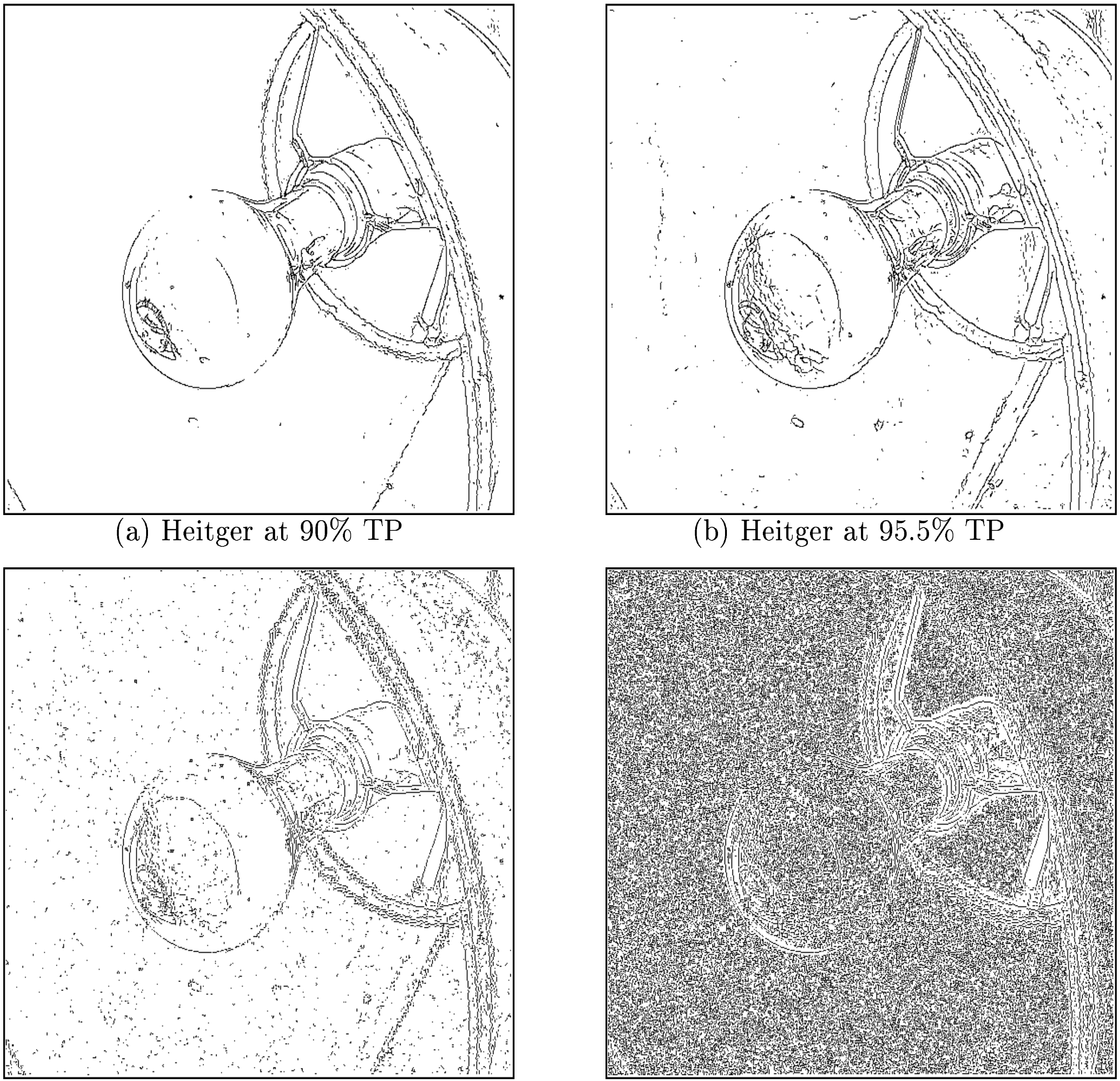


Figure 3: Stages in the Adaptive Sampling of Parameter Space.

5 Results

To have a better frame of reference for interpreting the meaning of different points on the ROC curves, it is useful to consider some example edge maps. Figure 4 presents sample edge maps for two different ROC points for the Heitger and Sobel detectors. Sample edge maps are presented for each detector at approximately 10% and 4.5% Unmatched GT (90% and 95.5% TP, respectively). The Heitger edge maps represent (10.5% Unmatched GT, 0.04% FP) and


(a) Heitger at 90% TP

(b) Heitger at 95.5% TP

(e) Sobel at 90% TP

(f) Sobel at 95.5% TP

Figure 4: Sample Edge Maps for Heitger and Sobel.

(4.6% Unmatched GT, 0.6% FP). The Sobel edge maps represent (9.9% Unmatched GT, 1.5% FP) and (4.6% Unmatched GT, 34.9% FP). At 10% Unmatched GT both detectors produce edge maps that seem subjectively reasonable, although the Heitger edge map is obviously better. The edge map for each detector gets subjectively worse in going from 10% to 4.5% Unmatched GT, due to the presence of more FP edges. However, the Heitger edge map still seems to be of reasonably quality, whereas the Sobel seems to have deteriorated to the point of becoming useless.

It is necessary to look at the experimental results from several viewpoints in order to make carefully considered conclusions. One view of the results is to compare the aggregate train and test ROC curves for the detectors. In evaluating these curves, it is important to also consider the numbers of images on which each detector was able to match the highest percentages of GT. In understanding the significance and repeatability of the differences between edge detectors, it is useful to compare the area under the training ROC curves across the individual images. Lastly, the relative computing time required by the detectors may be a useful secondary consideration. The next subsections present these various results.

5.1 Aggregate Train and Test Curves

The aggregate training curve represents the average performance of a detector when it is carefully tuned on each individual image. The aggregate training ROC curve for the fifty object images is obtained by averaging data from the fifty individual training curves. The aggregate training curve for the ten aerial images is obtained by averaging data from ten individual training ROC curves. These aggregate training curves are presented in Figure 5. Because of the number of detectors evaluated, they are presented in two groups. Group I is the Canny detector plus the five others whose ROC curve is similar to or slighter better than that of the Canny. Group II is the Canny detector plus the five others whose ROC curve is not as good as that of the Canny. It is apparent that the relative ranking between detectors is much the same across the object and aerial datasets.

The aggregate training curves represent an ideal level of image-specific tuning. Aggregate test curves would better represent the expected performance in most applications. The aggregate test ROC curve represents the average performance that can be expected from a detector after it is carefully tuned on one image and then applied across a set of images from the same domain. The aggregate test ROC curves are presented in Figure 6. The aggregate test curves again show that there is little difference in the relative ranking of the detectors across the two datasets. Several of the detectors appear to offer some improvement over the Canny, primarily at the highest values of % GT matched. However, recall that these are aggregate curves. Some of the detectors do not achieve the highest % GT matched on all images, as shown in Figure 7 below.

There is essentially no difference in the relative ranking of detectors from the training curves to the test curves. This may seem unusual. However, recall that for the one-parameter detectors there will be essentially no difference between their training and their test curves. Thus the differences that occur between the training and test comparisons are mainly that the performance of multi-parameter detectors (e.g., the Canny) loses ground to the one-parameter detectors (e.g., the Black). While this does occur, the effect is generally not large enough to change the relative ranking of detectors.

5.2 Frequency of Matching Highest Percent of GT

As mentioned earlier, the aggregate ROC curve for a detector is generally an average over a different number of images at different levels of % GT matched. This is because the detectors vary in the maximum percent of the GT edges that they can find in an image. Thus it is important to know which detectors could most frequently achieve the highest levels of % GT matched. The graphs in Figure 7 summarize this information. The Y axis represents the number of images, and the X axis represents the highest level of GT matched. Ideal performance on this measure would be indicated by a straight line across the top of the plot. Several points are apparent from looking at the graphs. One, there is again little difference in the relative rankings of the detectors between the object image dataset and the aerial image dataset. Two, we can tell that all of the detectors are able to match at least 84% of the specified GT edges at some level of FPs. We can also tell that the Sobel had the greatest trouble matching high levels of % GT. Lastly, the Heitger was able to reach the highest level of GT matched (99.75%) on every one of the images.

Considering the aggregate test ROC curves in the context of the % GT matched curves, it seems clear that the Heitger outperforms the other detectors at the highest levels of % GT matched. Also, some of the other detectors' apparent advantage over the Canny in the aggregated ROC curves is seen to be based on averages over different sets of images.

5.3 Test for Statistical Significance

Formulating a test for statistical significance of differences between algorithms based on ROC curves requires care. The Area Under the ROC Curve (AUC) is the traditional metric for comparison. However, since our ROC curves result from a parameter-training process, it is not clear that the assumptions underlying traditional AUC statistical tests are valid. We formulate a simple statistical test based on computing the AUC over a standardized range of % Unmatched GT for all detectors and images, and comparing the frequency with which one detector obtains a lower AUC than another.

One issue that arises in computing the AUCs is the exact range of % Unmatched GT to be used. The results presented here use the range of 0.25% to 25%. There is little reason to use a broader range than this because most detectors and images cannot match more than 99.75% of the GT at any level of FP, and most can match up to 75% of the GT edges at nearly 0% FP.

Another issue that arises is how to artificially extend the ROC curve in the case where it does not naturally reach 0.25% Unmatched GT. Let the upper-leftmost point on the ROC curve be (TP=X,FP=Y). Then add the points (TP=X,FP=25%) and (TP=0.25%,FP=25%) to finish the curve. The FP level of 25% for the extension is somewhat arbitrary. It needs to be high enough that the detector is effectively penalized for not naturally detecting the higher levels of GT, but if it is too high then the extension area dominates the comparison. Given that single-pixel-wide edges are used, at most one half of the pixels in a no-edge region could be labeled as edge pixels. This implies a theoretical limit of FP = 0.5, but in unusual cases, the implementations of some detectors do allow FP to exceed 0.5.

Using the AUCs for a set of images, we formulate the statistical test as follows. The sign test can be used to check for statistical significance without requiring the assumption that the differences between two detectors' AUCs follow a normal distribution [5]. The null hypothesis is that detector A is equally likely to have a higher or a lower AUC than detector B when both are trained on the same image. Under this assumption, there is less than a 1 in 2^N chance that detector A would generate a lower AUC than detector B on each of a set of N images. The expected number of times for detector A to yield a lower AUC than detector B on the set of 50 images is 25. The variance is 12.5 and the standard deviation is 3.54. Thus a value outside the range of 17 to 33 (two standard deviations from the mean) is sufficient evidence to reject the null hypothesis at the 0.05 level. For a set of 10 images, a value outside the range of 2 to 8 is evidence of a statistically significant result.

Data for this statistical test is presented in Table 3 separately for the fifty object images and the ten aerial images. Obviously most of the entries in these tables represent statistically significant results. An example statement that could be made from these results would be – *the Heitger detector, in comparison to any of the other detectors except for the Iverson on the aerial images, produces a better AUC (computed as defined here) for a statistically significant percentage of the images.* Note that the pattern of relative performance between detectors here largely agrees with the subjective impression obtained from the plots of aggregate ROC curves. Note also that the pattern of relative performance does not vary significantly with the type of imagery.

5.4 Speed, Parameter Sampling and Number of ROC Points

We observed a difference of about 100 to 1 in execution times of the detectors. (Because the Ahuja detector was not run at USF, we do not have estimates of execution time for it.) The fastest detectors were the Sobel detector and Smith and Brady's SUSAN detector. The slowest detectors were Black's robust anisotropic diffusion detector and Iverson and Zucker's

LogLin detector. However, there is no reason to believe that all of the implementations were created with equal attention to efficiency, and so the relative execution times provide only general information.

We also observed a factor of about 100 to 1 between the minimum and maximum number of points examined in the the adaptive search of the detectors' parameter spaces. See Table 4 for the numbers. There appears to be no strong correlation between this index and the ranking of detector performance.

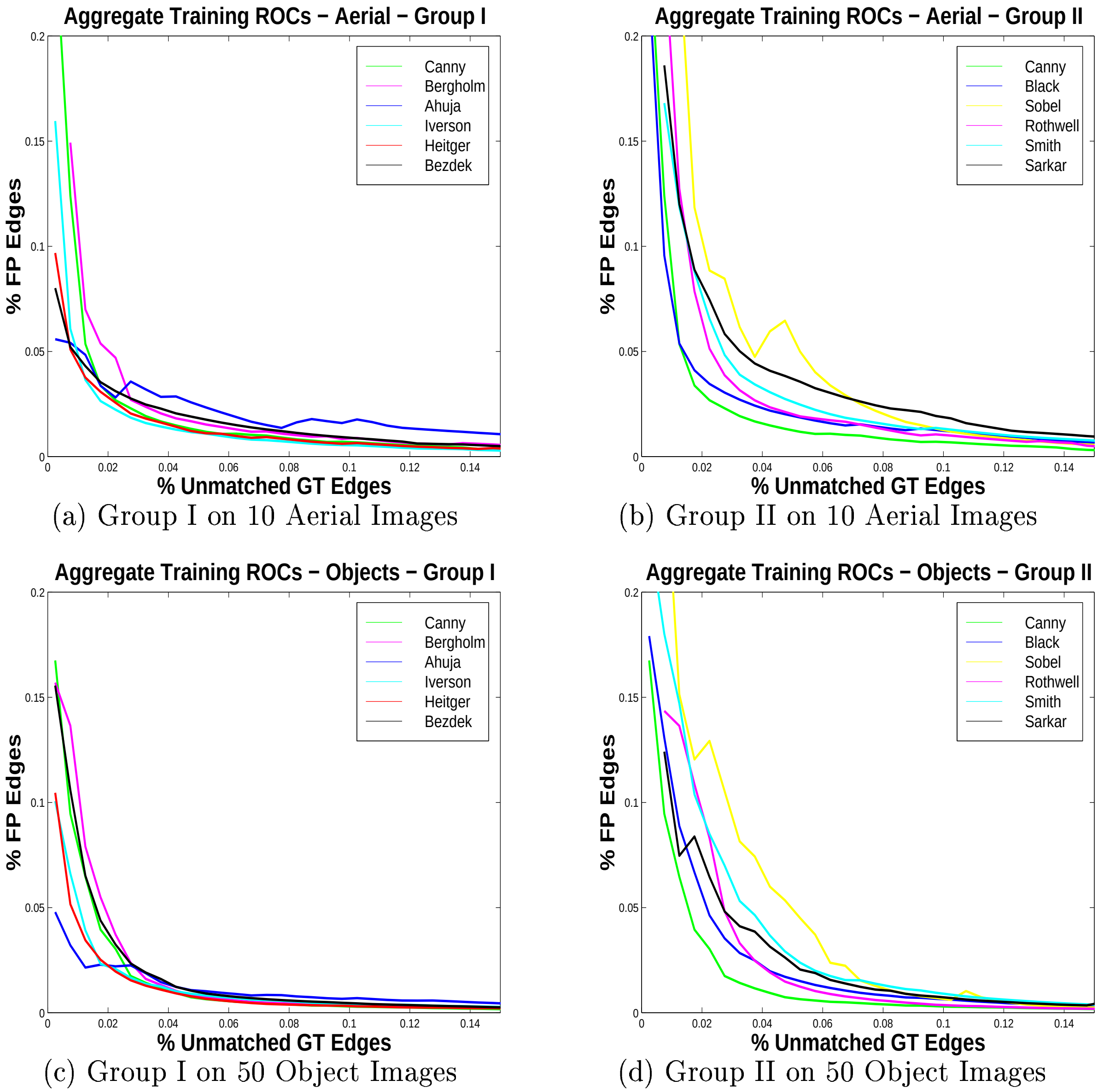
There is a modest factor of approximately 3 to 1 between the minimum and maximum average number of points on an ROC curve (other than the Ahuja detector). Again, there is no clear correlation to performance rank.

5.5 Similarity of Detectors' Edge Maps

It is potentially valuable to consider the GT found or missed in common by different detectors. This could reveal that some detectors make complementary mistakes, or point to situations in which all detectors fail. Consider results from detectors A and B at the points nearest to 20% Unmatched GT on their respective ROC curves. Each detector matches some 80% of the GT edges. Therefore 60% of GT edges must be matched by both detectors. On the remaining 40%, the two detectors may range from complete agreement to complete disagreement. For any pair of detectors, we can calculate the percent of the GT pixels not required to be matched by both detectors that actually are matched by both. This can vary between 0% and 100%, and is an indication of the degree to which the detectors naturally find the same edges. Tables 5 and 6 summarize this data for the two sets of images.

These tables suggest that there is fairly high agreement between detectors in terms of the GT edges matched. This argues against there being a large advantage to combining multiple of these detectors into a hybrid detector. A visual depiction of this point is given by Figure 8. This figure shows the result of a logical AND of the GT matched by ten of the detectors (the Ahuja is not included) at about 80% GT Matched for the airplane image. Figure 9 focuses on two detectors, the Canny and Heitger, and shows a GT edge map coded by color. Yellow represents

GT edge pixels matched by both detectors, blue represents GT matched only by Heitger, red represents GT matched only by Canny and black represents GT not matched by either. From initial inspection of such images such as those in Figures 8 and 9, there does not appear to be a common theme to the GT edges matched by one detector but not the other.



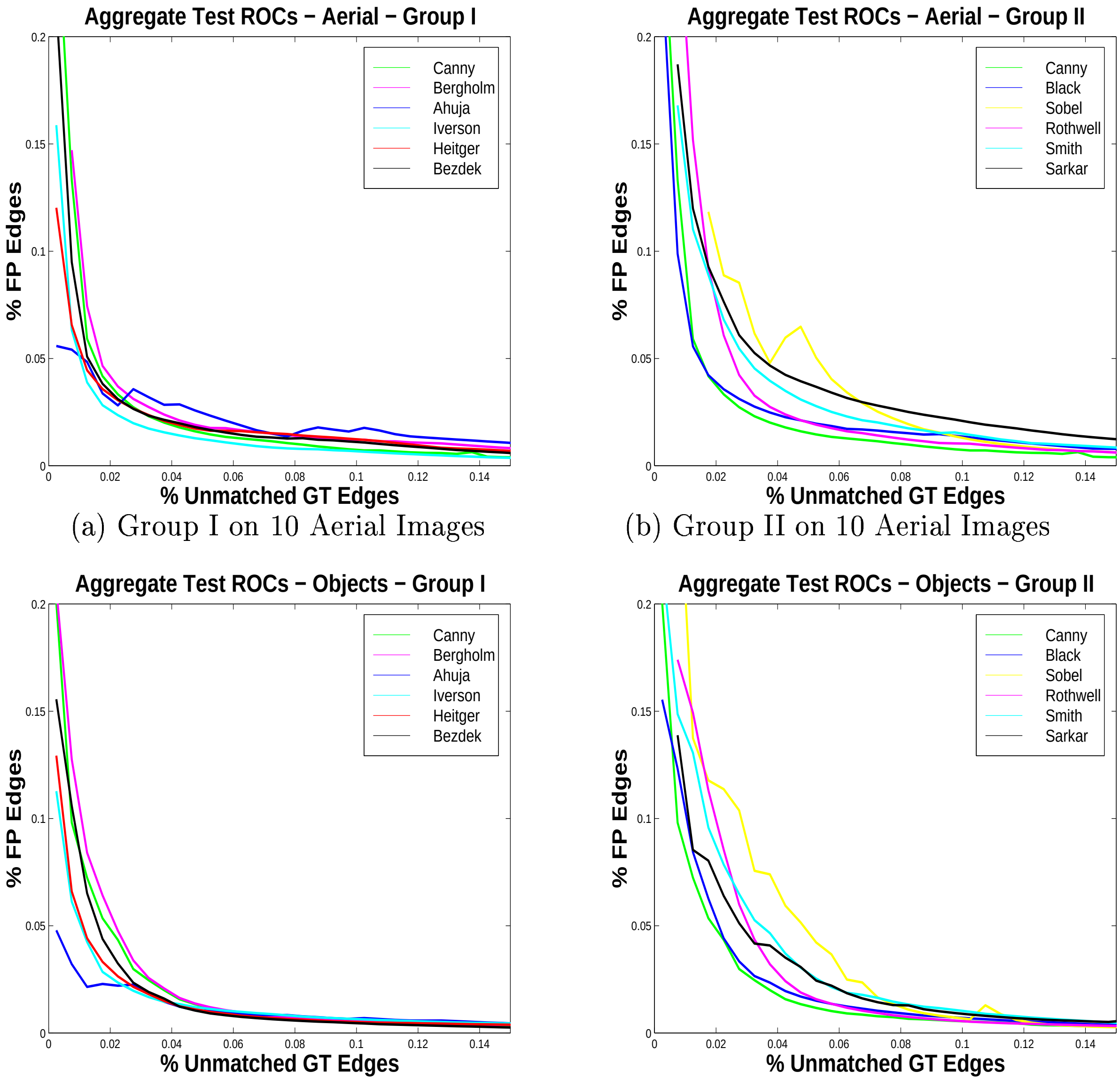
(a) Group I on 10 Aerial Images

(b) Group II on 10 Aerial Images

(c) Group I on 50 Object Images

(d) Group II on 50 Object Images

Figure 5: Aggregate Training ROC Curves for Object and Aerial Datasets.



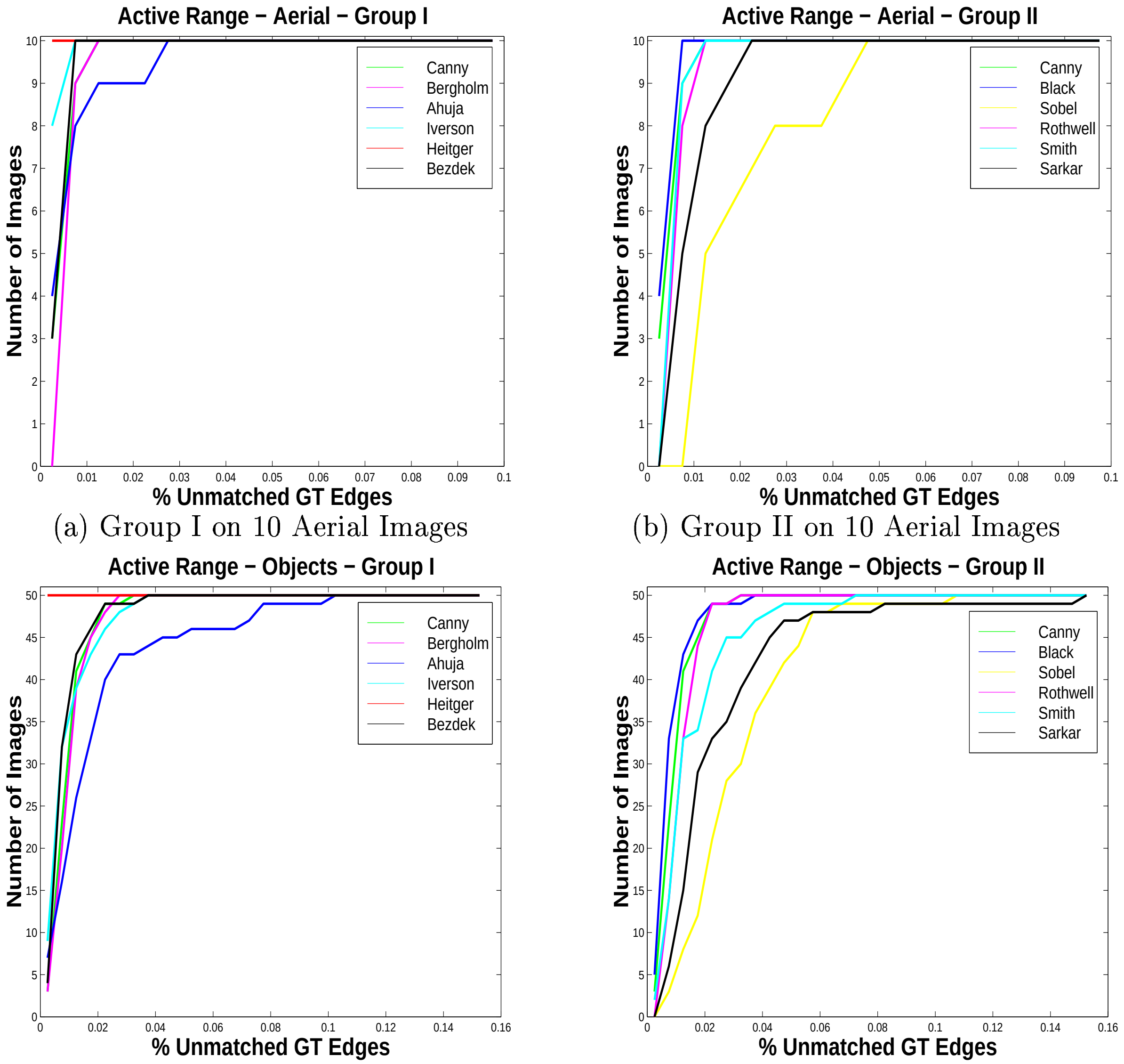
(a) Group I on 10 Aerial Images

(b) Group II on 10 Aerial Images

(c) Group I on 50 Object Images

(d) Group II on 50 Object Images

Figure 6: Aggregate Test ROC Curves for Object and Aerial Datasets.



(a) Group I on 10 Aerial Images

(b) Group II on 10 Aerial Images

(c) Group I on 50 Object Images

(d) Group II on 50 Object Images

Figure 7: Frequency of Levels of GT Matched for Object and Aerial Datasets.

Results for the Dataset of 50 Object Images

	Heit	Iver	Cann	Bezd	Berg	Blac	Roth	Sobe	Smit	Sark
Heitger	*									
Iverson	46	*								
Canny	46	31	*							
Bezdek	49	29	24	*						
Bergholm	49	36	39	39	*					
Black	50	40	34	35	24	*				
Rothwell	50	45	49	48	49	40	*			
Sobel	50	47	49	49	48	44	45	*		
Smith	50	49	50	50	49	48	40	9	*	
Sarkar	50	48	50	47	48	43	39	9	30	*
Ahuja	45	38	33	34	32	31	22	10	16	12

Results for the Dataset of 10 Aerial Images

	Heit	Iver	Cann	Bezd	Berg	Blac	Roth	Sobe	Smit	Sark
Heitger	*									
Iverson	5	*								
Canny	9	10	*							
Bezdek	9	8	4	*						
Bergholm	10	10	8	8	*					
Black	10	8	4	5	2	*				
Rothwell	10	10	10	10	10	9	*			
Sobel	10	10	10	10	10	9	8	*		
Smith	10	10	10	10	10	9	4	2	*	
Sarkar	10	10	10	10	10	9	10	6	7	*
Ahuja	9	7	6	6	5	6	3	1	1	0

Table 3: Relative AUC Comparison Across Training ROCs.
Entries in the tables indicate the number of times that the column-named detector had an AUC better than the row-named detector.

Detector	average # of sample points in training	average # of points on ROC
Canny	1439	38
Bergholm	3783	46
Sarkar	1317	43
Rothwell	4261	50
Heitger	2902	55
Sobel	248	40
Black	68	34
Bezdek	833	46
Iversion	2005	47
Smith	39	17
Ahuja	9	9

Table 4: Number of Parameter Settings and Points on ROC Curve.

	Berg	Cann	Heit	Blac	Roth	Sobe	Smit	Sark	Iver	Bezd
Bergholm	*	–	–	–	–	–	–	–	–	–
Canny	73	*	–	–	–	–	–	–	–	–
Heitger	78	74	*	–	–	–	–	–	–	–
Black	76	72	78	*	–	–	–	–	–	–
Rothwell	72	69	74	71	*	–	–	–	–	–
Sobel	64	61	65	68	67	*	–	–	–	–
Smith	66	63	69	73	69	68	*	–	–	–
Sarkar	66	59	60	59	60	54	50	–	–	–
Iverson	76	63	68	69	64	63	57	57	*	–
Bezdek	66	72	77	80	73	68	65	58	75	*
Ahuja	67	66	68	67	62	54	52	53	55	62

Table 5: % Free Agreement In Matched GT Edge Pixels: 50 Object Images.

	Berg	Cann	Heit	Blac	Roth	Sobe	Smit	Sark	Iver	Bezd
Bergholm	*	–	–	–	–	–	–	–	–	–
Canny	78	*	–	–	–	–	–	–	–	–
Heitger	83	79	*	–	–	–	–	–	–	–
Black	79	74	70	*	–	–	–	–	–	–
Rothwell	72	71	66	73	*	–	–	–	–	–
Sobel	67	64	59	70	69	*	–	–	–	–
Smith	73	68	64	81	74	72	*	–	–	–
Sarkar	61	63	56	60	64	57	54	*	–	–
Iverson	71	66	65	76	69	69	68	59	*	–
Bezdek	78	72	68	83	74	71	74	58	79	*
Ahuja	67	66	68	67	62	54	52	53	55	62

Table 6: % Free Agreement In Matched GT Edge Pixels: 10 Aerial Images.

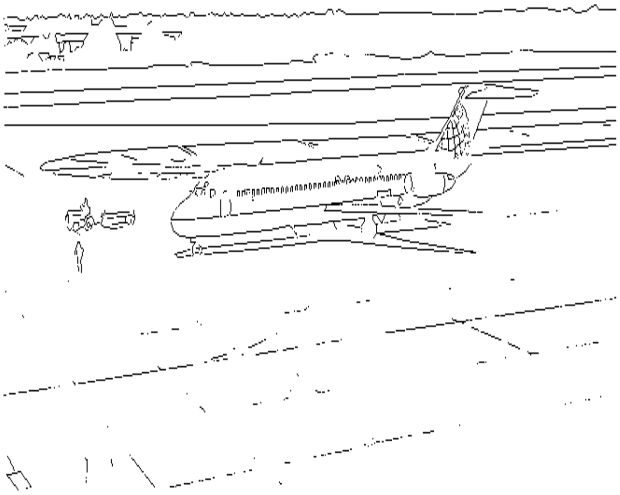


Figure 8: GT Edges Matched By 10/10 Detectors at 80% GT Matched.

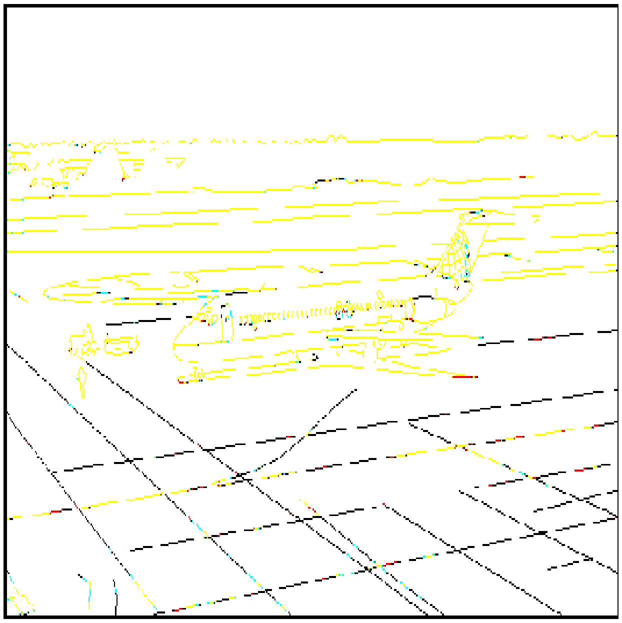


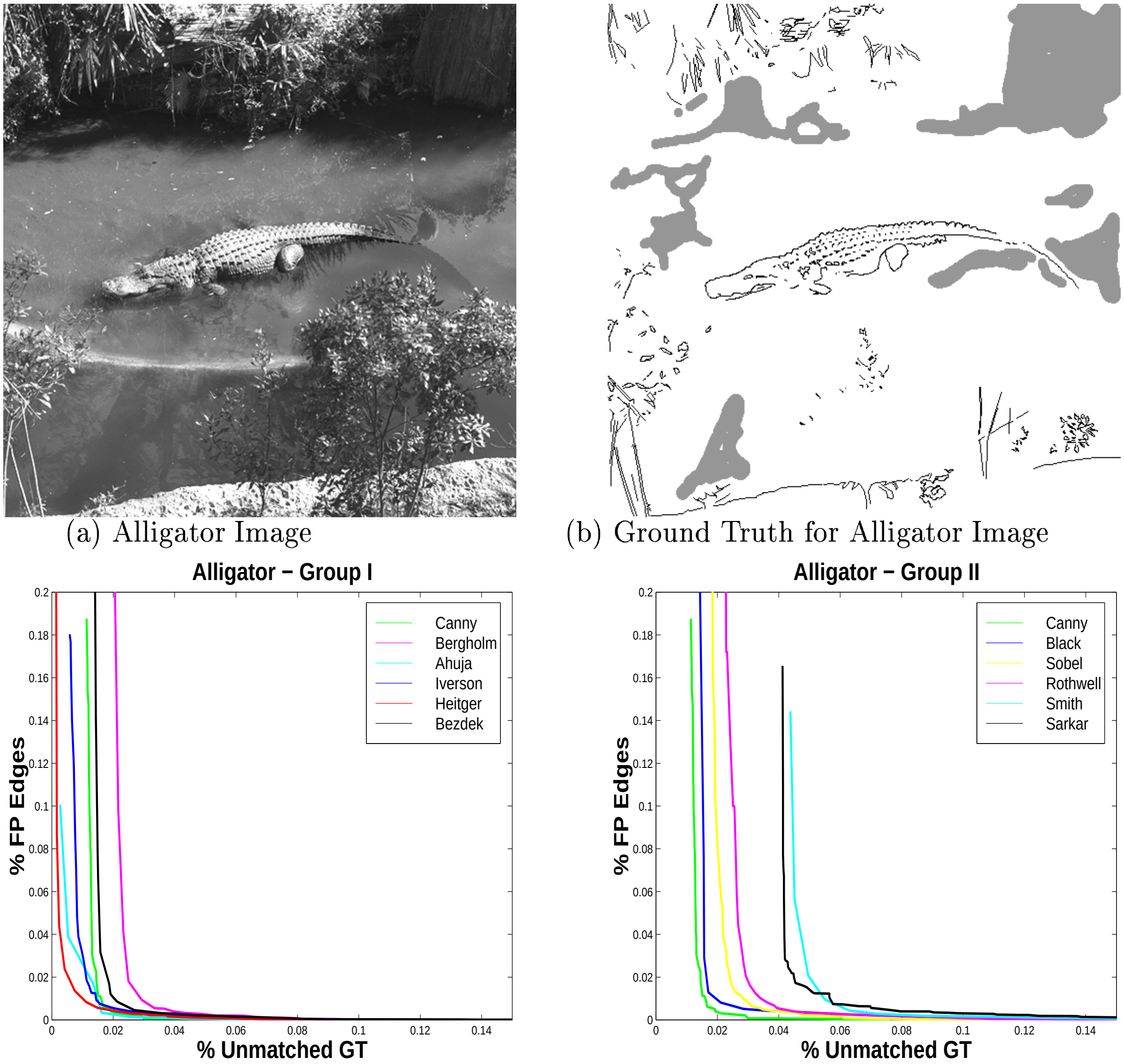
Figure 9: GT edge pixels matched by Canny/Heitger.

6 Summary and Discussion

We have demonstrated a framework for evaluation of edge detector performance using empirical ROC curves. The framework uses real images, manually-specified ground truth, adaptive sampling of edge detector parameter space, and a train-and-test paradigm. Results are presented for eleven detectors using a set of fifty object images and a set of ten aerial images. Results include aggregate train and test ROC curves, the percent of images for which the highest level of GT edges was matched, and the percent of images for which better AUCs were obtained.

It is important to consider that substantial differences in edge detector performance emerge only at higher TP levels. Consider an application that requires matching only 75% of the GT edges. This can be achieved at a level of 0% FPs, as FPs are counted in this framework, by almost any of the detectors on almost all of the images. However, if an application requires matching 90% or greater of the GT edges, there are large differences between detectors.

Overall, the Heitger detector appears to offer the highest level of performance. The relative ranking of detectors appears to be insensitive to minor details of the GT specification, provided that there are substantial amounts of both edge and no-edge in the GT. The relative ranking of detectors is also stable across images and type of imagery. Results were shown for object imagery and aerial imagery. The subject matter of the object imagery could be subdivided into indoor/outdoor or man-made/natural and the detector performance rankings would remain essentially the same. The highest- and lowest-ranked detectors are the most stable, with the order of the middle-ranked detectors interchanged more frequently. As an example, one of the more unusual images and set of ROC curve results is shown in Figure 10. The image is of a natural, outdoor scene containing an alligator. It can be challenging to specify the GT for this type of image. Note the that Sobel detector does not result in the worst ROC curve for this image. Instead, it ranks ahead of several other detectors. However, the order of the highest ranking detectors is still typical of general result. Overall, the results suggest that selection of a “best” edge detector can be a fairly general decision.



(a) Alligator Image

(b) Ground Truth for Alligator Image

(a) Group I on Alligator Image

(b) Group II on Alligator Image

Figure 10: Training ROC Curves for Example Natural, Outdoor Scene.

6.1 Comparison to Results of Forbes and Draper

Forbes and Draper have presented results of a similar ROC-style approach to edge detector evaluation [18]. They used their own implementation of an ROC framework that follows our approach as described earlier [6, 12]. However, they use simulated images in order to explore the effects of parameters of the image acquisition process. Because they used simulated images, they were also able to use an edge / no-edge GT and avoid the don't-care marking used in our ground truth. An example image and its ground truth from [18] are shown in Figure 11.

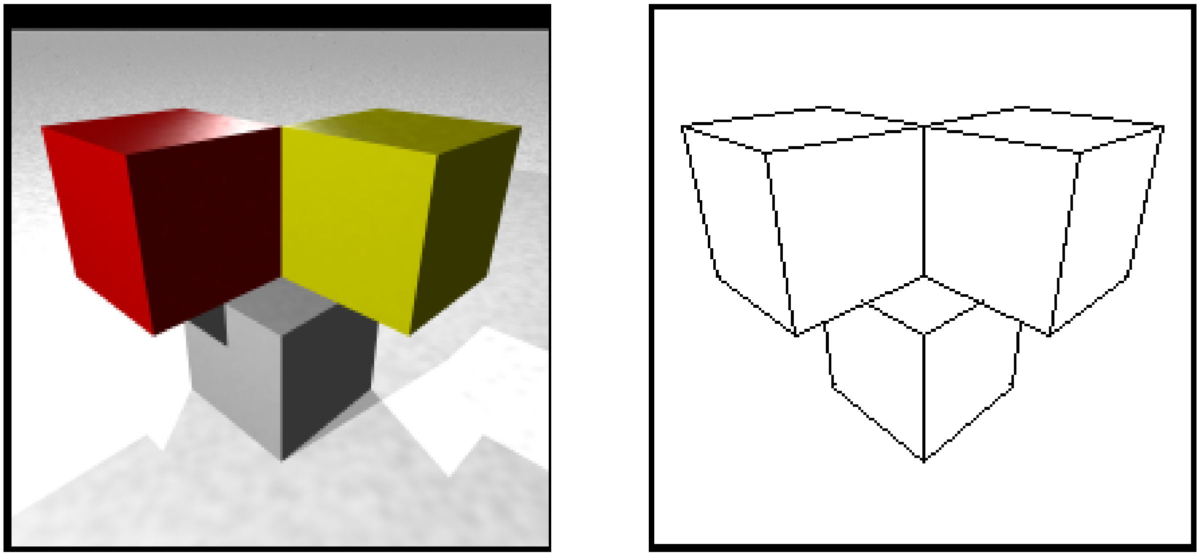


Figure 11: Example synthetic image and its GT from [18].

Forbes and Draper report results that differ from ours on several important points [18]. Compared to our results, they report that (1) relative rankings of edge detectors are not nearly as consistent as we found, (2) Smith and Brady's SUSAN detector generally achieves better performance relative to other detectors than we found, (3) extreme parameter settings are much more often selected for use in the Canny edge detector in their framework, and (4) performance of an edge detector can vary substantially with factors such as the resolution of the image. Our two research groups pursued substantial correspondence and collaboration in order to identify the cause(s) of the disparate results. Several factors were identified.

One factor is that we used the SUSAN detector with its optional smaller (3×3) mask, whereas the results in [18] use the default 37-pixel circular mask embedded in a 7×7 window. This mask leaves a larger border around the image boundary in which no edge pixels are detected. When TP/FP results are counted over the whole image, with the two masks detecting edges over different subsets of the whole image, the larger mask generally gives better performance. This is because its performance is effectively boosted by being unable to report false positives for a larger border around the image. When TP/FP results are counted over the same image area, for which both masks report edge detection results, the smaller mask generally gives better performance.

Another factor is that our parameter search processes differed in one detail. Our framework requires that the search go at least two steps. That is, the search cannot stop at a $4 \times 4 \times 7$ step, but must explore at least one further step. (This implementation detail was not reported in the description of our framework in [6, 12].) Forcing the search process to take at least two iterations gives better ROC curves for some images.

Several additional factors center on the nature of the synthetic images in [18]. The image / ground truth pair in Figure 11 show several properties that make it special relative to the real image data used in our study. One property is the simple foreground / background nature of the scene, resulting in all of the GT edges forming one connected group. In principle, a Canny-like hysteresis procedure could start with just the one highest-strength edge pixel and still trace out edges to match all of the ground truth. Thus a value of 1.0 for the high hysteresis threshold in the Canny detector becomes a plausible choice. Another property is that there are relatively few GT edge pixels. The average number of GT edge pixels in this sequence of images in Forbes and Draper study is approximately two orders of magnitude less than in our real image data. Another special property lies in the approach to marking GT edges. As shown in Figure 11, their GT contains only edges that are due to object geometry. It does not include edges due to lighting and shadows, even though the strength of some lighting/shadow edges in the image may be stronger than that of some object edges. Thus the GT mixes the question of distinguishing the cause of the edge (object geometry versus lighting) with the presence of an edge. All of the detectors evaluated were designed simply to detect edges, not to distinguish their cause.

Forbes and Draper have run a modified version of their implementation on a more complex synthetic image, without shadow edges, and using the smaller mask for the SUSAN detector. In this limited experiment their results appear to largely agree with ours, in the sense that their method produces stable rankings of detectors that match the rankings produced by our method. We still favor the use of real images and the three-valued ground truth, as we feel that it more explicitly forces one to confront the messy issues that are essential to the utility of the results. However, it appears possible that the synthetic-image approach of Forbes and Draper could reasonably be used to make valid evaluations of detector performance.

6.2 Relative Importance of Kernel and Post-Processing

One element of folklore in the edge detection community is that the traditional Sobel detector augmented with the non-maxima suppression and hysteresis of a Canny detector would perform about the same as the Canny detector. Our results show that this is not true. Our implementations of the Sobel and the Canny use the same non-maxima suppression and hysteresis routines, and the Canny performs much better. Figure 12 shows sample edge maps for the two detectors using comparable parameter settings. For both detectors, a low hysteresis threshold of 0.5 and a high hysteresis threshold of 0.94 are used. For the Canny, the σ value used is 0.5, roughly corresponding to the 3×3 mask size of the Sobel. Note that the Canny edges appear subjectively to be better organized into smooth, continuous contours. This suggests that the smoothing step and / or the particular filter shape are important in addition to the non-maxima suppression and hysteresis routines.

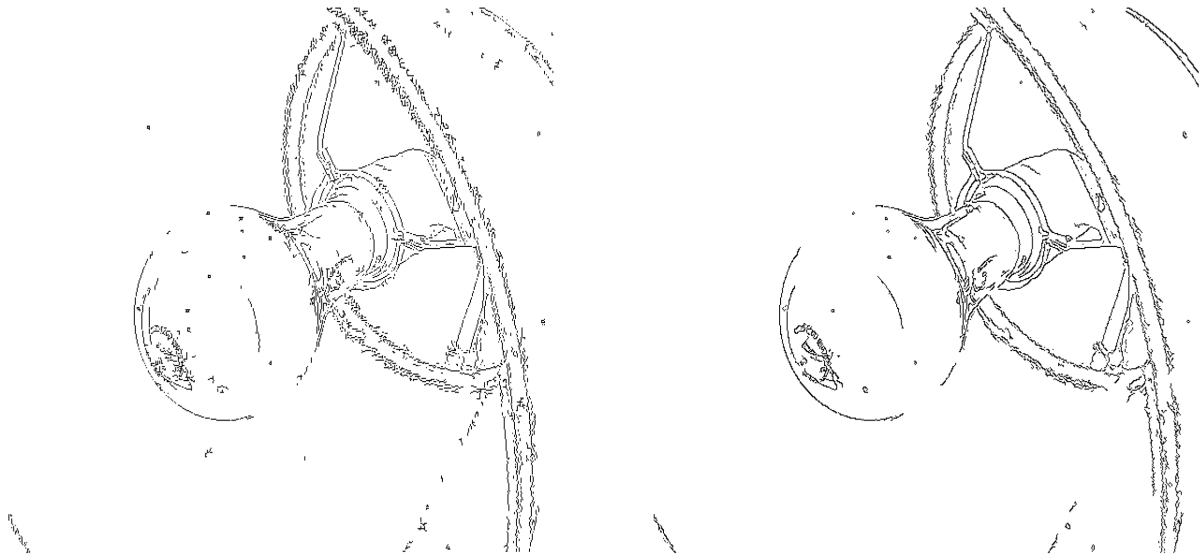


Figure 12: Example corresponding edge maps from Sobel (left) and Canny (right).

6.3 Suggested Use of This Framework

Evaluation of a proposed new edge detector could be done using all of the sixty images, and using the full train-and-test methodology. However, considering the results of this evaluation, it seems that a meaningful performance comparison can generally be done at much less computational cost. A comparison of the aggregating training ROC curves and the AUCs for a subset of the images should be sufficient in most cases. A full train-and-test evaluation might be more appropriate for a detector with a large number of parameters. A larger number of images might be more appropriate if it was important to determine statistical significance of similarly-performing detectors.

The complete set of images, ground truth, and software for creating ROC curves is contained in the tar file available from our web site. The implementation also includes scripts to compare a proposed new detector to the Heitger based on a subset of the images.

Acknowledgments

This work is supported by National Science Foundation grants IIS-9731821 and EIA-9729904. Earlier results in this line of work were presented at the 1999 *Computer Vision and Pattern Recognition* conference [6] and the 1998 *Workshop on Empirical Evaluation of Computer Vision Algorithms* [12].

The authors would like to thank all of the researchers who made their edge detector implementations available to the community. We would especially like to thank Michael Black for responding to a number of questions about his work, and Kar-Han Tan and Narendra Ahuja for processing image files and answering numerous questions about their implementation. We would also especially like to thank Lance Forbes and Bruce Draper for extensive correspondence regarding the details of their work.

References

[1] I. E. Abdou and W. K. Pratt, Quantitative design and evaluation of enhancement/thresholding edge detectors, *Proceedings of the IEEE* 67 (5), 753-763, 1979.

[2] F. Bergholm, Edge Focusing, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 9, 726-741, 1987.

[3] J.C. Bezdek, R. Chandrasekhar, and Y. Attikiouzel, A geometric approach to edge detection, *IEEE Transactions on Fuzzy Systems* 6 (1), 52-75, 1998.

[4] M.J. Black *et al.*, Robust anisotropic diffusion, *IEEE Transactions on Image Processing* 7 (3), 421-432, 1998.

[5] M. Bland, Medical Statistics, oxford University Press, Oxford, 1995.

[6] K.W. Bowyer, C. Kranenburg and S. Dougherty. Edge detector evaluation using empirical ROC curves, *Computer Vision and Pattern Recognition* (CVPR '99), Fort Collins, Colorado (June 1999), I:354-359.

[7] K.W. Bowyer and P.J. Phillips, editors, *Empirical Evaluation Techniques in Computer Vision*, IEEE Computer Society Press, 1998.

[8] D. J. Bryant and D. W. Bouldin, Evaluation of edge operators using relative and absolute grading, *IEEE Conf. on Pattern Recognition and Image Processing*, pp. 138–145, 1979.

[9] J. Canny, A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8, 679-698, 1986,

[10] K. Cho, P. Meer and J. Cabrera, Quantitative evaluation of performance through bootstrapping: edge detection, *Int. Symp. on Computer Vision*, 491-496, 1995.

[11] H. Christensen and W. Foerstner, guest editors, Special issue on performance evaluation, *Machine Vision and Applications* 9 (5), 1997.

[12] S. Dougherty and K.W. Bowyer, K.W. Objective evaluation of edge detectors using a formally defined framework, 211-234 in [7].

[13] E. S. Deutsch and J. R. Fram, A quantitative study of the orientation bias of some edge detector schemes, *IEEE Transactions on Computers* 27 (3), pp. 205-213, 1978.

[14] P.W. Eichel and E.J. Delp, Quantitative analysis of a moment-based edge operator, *IEEE Transactions on Systems, Man, and Cybernetics* 20 (1), pp. 59-66, 1990.

[15] W.E.L. Grimson and E.C. Hildreth, Comments on "Digital step edges from zero crossings of second directional derivatives," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 7 (1), 121-127, January 1985.

[16] R.M. Haralick, Digital step edges from zero crossings of second directional derivatives, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6 (1), 58-68, January 1984.

[17] R.M. Haralick, Author's reply, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 7 (1), 127-129, January 1985.

[18] L.A. Forbes and B.A. Draper, Inconsistencies in edge detector evaluation, *Computer Vision and Pattern Recognition* (CVPR '00), Hilton Head, SC (June 2000), 398-404.

[19] M. Heath *et al.*, A Robust Visual Method for Assessing the Relative Performance of Edge Detection Algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19 (12), 1338-1359, December 1997.

[20] F. Heitger, Feature Detection using Suppression and Enhancement, TR 163, Image Science Lab, ETH-Zurich, 1995.

[21] L.A. Iverson and S.W. Zucker, Logical/linear operators for image curves, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17 (10), 982-996, 1995.

[22] X.Y. Jiang *et al.*, A methodology for evaluating edge detection techniques for range images, *1995 Asian Conference on Computer Vision*, volume II, 415-419.

[23] T. Kanungo, M. Y. Jaisimha, and R. M. Haralick, A methodology for quantitative performance evaluation of detection algorithms, *IEEE Transactions on Image Processing* 4, pp. 1667–1674, Dec. 1995.

[24] L. Kitchen and A. Rosenfeld, Edge evaluation using local edge coherence, *IEEE Transactions SMC* 11 (9), 597-605, 1981.

[25] W.H.H.J. Lunscher and M.P. Beddoes, Optimal edge detector evaluation, *IEEE Transactions SMC* 16 (2), pp. 304–312, 1986.

[26] G. Liu and R.M. Haralick, Assignment problem in edge detector performance evaluation, *Computer Vision and Pattern Recognition* (CVPR '00), volume 1, 26-31.

[27] P. L. Palmer, H. Dabis and J. Kittler, A Performance Measure for Boundary Detection Algorithms, *Computer Vision and Image Understanding* 63 (3), pp. 476–494, 1996.

[28] V. Ramesh and R. M. Haralick, Performance characterization of edge detectors, *SPIE Vol. 1708 Applications of AI X* , pp. 252–266, 1992.

[29] C. Rothwell *et al.*, Driving Vision by Topology, *IEEE Int. Symp. on Computer Vision,* 395-400, 1995.

[30] M. Salotti *et al.*, Evaluation of Edge Detectors: Critics and Proposal, *Workshop on Performance Characterization of Vision Algorithms*, 1996.

[31] Shin, M., Goldgof, D.B., and Bowyer, K.W. An objective comparison methodology of edge detection algorithms using a structure from motion task, *Computer Vision and Pattern Recognition* (CVPR '98), 190-195.

[32] Shin, M.C, Bowyer, K.W. and Goldgof, D.B. Comparison of edge detectors using an object recognition task, *Computer Vision and Pattern Recognition* (CVPR '99), Fort Collins, Colorado (June 1999), I:360-365.

[33] S. Sarkar and K. Boyer, Optimal Infinite Impulse Response Zero-Crossing Based Edge Detection, *Computer Vision, Graphics, and Image Processing* 54 (2), 224-243, 1991.

[34] S. Smith and M. Brady, SUSAN - a new approach to low level image processing, *International Journal of Computer Vision* 23 (1), 45-78, 1997.

[35] L.J. Spreeuwers and F. van der Heijden, Evaluation of edge detectors using average risk, *International Conference on Pattern Recognition*, the Netherlands, pp. 771-774 of volume 3, 1992.

[36] R. N. Strickland and D. K. Cheng, Adaptable edge quality metric, *Optical Engineering* 32 (5), pp. 944–951, 1993.

[37] M. Tabb and N. Ahuja, Multiscale image segmentation by integrated edge and region detection, *IEEE Transactions on Image Processing* 6 (5), 642-655, 1997.

[38] S. Venkatesh and L.J. Kitchen, Edge evaluation using necessary components, *CVGIP: Graphical Models and Image Processing* 54 (1), pp. 23-30, 1992.

[39] M.A. Viergever, H.S. Stiehl, R. Klette, and K. Vincken, *Performance Characterization and Evaluation of Computer Vision Algorithms*, Kluwer Academic Publishers, 2000.

[40] Y.T. Zhou, V. Venkateshwar, and R. Chellappa, Edge detection and linear feature extraction using a 2D random field model, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11, pp. 84–95, 1989.

[41] Q. Zhu, Efficient evaluations of edge connectivity and width uniformity, *Image and Vision Computing* 14, pp. 21–34, 1996.