An Integrated Gradient Edge Detector – Theory and Performance Evaluation

Visvanathan Ramesh *
Robert M Haralick
Intelligent Systems Laboratory, Dept. of EE, FT-10
University of Washington, Seattle WA 98195
e-mail:{rameshv,haralick}@george.ee.washington.edu

Abstract

Edge detection is a fundamental step in Computer Vision. Several edge detection schemes have been proposed in the computer vision literature. Some of the prominent ones include the Marr-Hildreth edge detector, the Haralick edge detector, and the Canny edge detector. Previous work by Ramesh and Haralick, ([9],[11]), and Wang and Binford [15] have provided theoretical and empirical evaluation of some of the edge detection schemes. This paper shows how we use the insights gained during the performance evaluation to develop a better edge detector. We illustrate that the precision of the edge orientation estimate is a function of the input signal to noise ratio (the ratio of the true gradient magnitude to the grey level noise standard deviation) and the neighborhood size used in the edge detector. This observation has direct impact on edge orientation estimation. We also illustrate that an appropriate measure for a pixel being an edge pixel along a given direction is the integrated gradient along that direction. We use the point that the minimum and maximum integrated gradient magnitudes are simultaneously high at edge locations to detect edge pixels. The paper also provides theoretical and empirical analysis of the performance of the operator.

1 Introduction

Edge detection is a fundamental step in Computer Vision. Several edge detection schemes have been proposed in the literature. Some of the prominent ones include the Marr-Hildreth edge detector, the Haralick edge detector, and the Canny edge detector. Previous work by Ramesh and Haralick, ([9],[11]), provided a theoretical and empirical evaluation of the performance of gradient based and morphological edge detection schemes. Wang and Binford [15] also have studied the performance of the Canny edge detector and have shown how one can automatically set tuning parameters for the Canny edge detector. A more recent paper by Ramesh et al [12] illustrates how one can set the tuning parameters of an image analysis sequence involving edge detection, corner finding and chain length thresholding. There are major differences in the theoretical analysis of the two papers, and the expression for the gradient threshold derived in Ramesh et al is a more general form incorporating prior knowledge. Incidentally, both the papers give almost identical expressions for the high gradient threshold used in the Canny edge detector. Ramesh et al also show how one can set the low threshold used in Canny's hysteresis linking procedure. While that analysis illustrates how it is possible to set thresholds to optimise a given criterion function, the analysis does not show how one can tackle the fundamental problems in edge parameter estimation. In this paper, we use the insights gained by systematic performance evaluation of edge detectors to derive a new edge detector. We will first illustrate that the edge orientation estimates have very large uncertainty when the signal to noise ratio for a given edge is low. It is shown that the precision of the edge orientation estimate is a function of the signal to noise ratio (the ratio of the gradient magnitude to the standard deviation of the noise at the output after applying the blurring filter). For example, if we assume a 5 by 5 box filter and a signal to noise ratio of 2, the size of the 95% confidence interval for the orientation estimate is approximately 24 degrees. Worse yet, if one uses a 3 by 3 kernel this confidence interval spans over 180 degrees. This observation has significant impact on how edge orientation estimation should be done. We also take a look at what is a good measure for an edge. Previous work has been done on facet model based integrated gradient edge extraction scheme by Zuniga and Haralick [16]. Our work complements their work. In this paper we describe a new edge detector based on integrated gradients. We derive our measure by viewing the problem from a totally different perspective. Our algorithm estimates the integrated gradient magnitude in several discrete directions. This is very similar to steerable filters developed by Perona et al [7]. A theoretical analysis of the edge detector performance is also performed. A preliminary empirical evaluation is given. We are in the process of comparing this scheme to conventional edge detectors.

*Funding for this work from ARPA Contract 92-F1426000-000 is gratefully acknowledged.
2 Motivation and Algorithm

In this section we discuss some of the fundamental problems with current edge finders and linkers. First, we illustrate the problem with the estimation scheme used to compute the edge orientation. We show that the edge orientation estimate has a significantly large variance when the true edge gradient is rather low. Then we discuss the problems that edge detectors have in complex scenes. Edge detectors typically have problems coping up with the shading in images. Since the edge scale in these complex images vary between region to region, the edge operators will have to be applied at different scales and the results from each scale have to be combined.

2.1 Estimation of Edge Orientation - Problems

One way of estimating the gradient magnitude in a given pixel’s neighborhood is by using the expression:

$$\hat{g} = \sqrt{\hat{\alpha}^2 + \hat{\beta}^2}$$

(1)

where $\hat{\alpha}$ and $\hat{\beta}$ are the coefficients of the best least squares fit plane $\alpha x + \beta y + \gamma$.

Let $K$ denote the width of the square neighborhood used in the gradient estimation step. Assuming that the individual pixel gray values are perturbed with i.i.d zero mean Gaussian random variables with standard deviation $\sigma$ we illustrate in [10] that the orientation estimate for an edge computed by the expression:

$$\hat{\theta} = \tan^{-1}(\hat{\beta}/\hat{\alpha})$$

(2)

is distributed as a Von Mises distribution. In fact, the conditional distribution for the orientation estimate given the estimated gradient and the true gradient value is:

$$P(\theta = \theta|\theta_0, \sigma, g, \hat{g}) = \frac{1}{2\pi I_0(\kappa)} \exp \{\kappa g e^{i(\theta - \theta_0)} - \pi \leq \theta < \pi \}$$

(3)

Here $\kappa$ is equal to: $g \Sigma \sigma^2/\sigma^2$, $\theta_0$ is the true orientation value and $I_0(\kappa)$ is the modified Bessel function of first kind and order 0. It is clear from the above expression that the orientation estimate has significantly high variance when the true gradient magnitude is low. It can be seen that as $g$ tends to infinity the precision parameter tends to infinity. As $g \to 0$, the distribution function approaches the uniform distribution. We see the following from the above expression:

- The precision of the orientation estimate is directly proportional to $\sigma$ and $\hat{g}$. An estimate for the precision of the direction estimate can be obtained by replacing $g$ with the estimated gradient $\hat{g}$. Thus, the estimate for precision of the orientation estimate is: $2\hat{g} \sigma^2$.

- The precision of the orientation estimate is a function of the neighborhood size. The larger the window size, the better the precision of the orientation estimate. For example, suppose that we have two edges with gradient magnitudes $g_1$ and $g_2$ respectively (assume: $g_2 < g_1$). Then in order to estimate the orientation of the second edge as precisely as the orientation of the first edge one will have to use a larger neighborhood size.

These results were seen in our experiments with the BURNS line finder. Specifically, we found that at low signal to noise ratios the line finder fragments the ideal line segment into small pieces. Even at low gradient thresholds the fragmentation was seen. This is mainly due to the fact that the variance in direction estimate was rather high at low signal to noise ratios. The Burns line finder first partitions the orientation space into a number of bins and computes the connected components of pixels with the same orientation labels. At low SNR’s the probability of misclassification is rather high because the orientation estimate has large variance.

2.2 Measure of Edge Strength

Previous edge detectors have estimated edge strength in a variety of ways. Some detectors use the height of the zero-crossing of the second directional derivative, while others use the isotropic gradient magnitude at a particular pixel. Some others have used non-linear schemes to estimate edge strength or contrast across the edge. Zuniga and Haralick have used the integrated gradient magnitude as a measure. We arrive at the conclusion that the integrated gradient magnitude along a given direction is a reliable measure of edge strength from a different perspective.

Let $G(r,c)$ denote the estimated gradient magnitude at image pixel location $(r,c)$. Imagine that we threshold the gradient image $G$ at a threshold $t$ to obtain a threshold image $\hat{G}$. Let $W(t, \theta, r, c)$ denote the cardinality of the set:

$$S = \{(x,y)|\hat{G}(x,y) = 1, (x,y) \in L(\theta,K,r,c)\}$$

(4)

Here $L(\theta,K,r,c)$ is the set of integer coordinates in the digital line neighborhood centered around $(r,c)$ with length $K$ and orientation $\theta$. $W(t, \theta, r, c)$ gives the number of pixels in the domain of the oriented digital line segment that are classified as edge pixels. An estimate of the probability of the current pixel being an edge pixel in direction $\theta$, $\hat{p}(r,c)(\text{edge}; t)$, is given by:

$$\hat{p}(r,c)(\text{edge}; t) = W(t, \theta, r, c)/K$$

(5)

Computing the integral:

$$\int_T T \hat{p}(r,c)(\text{edge}; t) dt$$

(6)

gives us a measure of the current pixel’s edge strength in a given direction. Although it seems like we will have to compute the weighted sum of the probability estimates by thresholding the gradient magnitude image at each possible threshold, we will now illustrate that this is not necessary. We show that the above equation is equivalent to the sum of the squared gradient magnitudes at the locations $(r,c)$’s in the discrete
line neighborhood. To see this, let \( \hat{g}(1), \hat{g}(2), \ldots, \hat{g}(K) \) denote the order statistics of the sequence of gradient magnitudes \( \hat{G}(x, y) \). Clearly:

\[
W(t, \theta, r, c) = \begin{cases} 
K & t \leq \hat{g}(1) \\
K - 1 & t \leq \hat{g}(2) \\
\vdots & \\
1 & t \leq \hat{g}(K) \\
0 & t > \hat{g}(K)
\end{cases}
\]

Thus equation 6 becomes:

\[
\int_0^{\hat{g}(1)} t dt + \frac{K - 1}{K} \int_{\hat{g}(1)}^{\hat{g}(2)} t dt + \cdots + \frac{1}{K} \int_{\hat{g}(K)}^{\hat{g}(K-1)} t dt.
\]

This is equal to the sum:

\[
\frac{1}{2K} \sum_{i=1}^{K} \hat{g}_i^2 = \frac{1}{2K} \sum_{(x,y) \in L(\theta, K, r, c)} \hat{G}^2(x, y).
\]

Other researchers have used the integrated gradients along a direction normal to the tangent direction at a point on the contour. Specifically, and Fua and Leclerc [2] have used this measure as one of the terms in their objective functions for an algorithm based on Snakes [14]. The main difference between their approach and ours is that they need an initial guess for the contour solution and we do not. A recent paper by Neunenschwander et al. [6] describes how an user could specify just the ending points of the curve and the curve could be obtained starting from this guess. Our algorithm does not need any user interaction and is motivated from a different perspective.

### 2.3 Basic Algorithm

We have seen that a good measure of edge strength is the integrated gradient magnitude. We have also seen that the orientation estimate computed by taking the inverse tangent of the ratio of column to row gradient magnitudes has a large variance when the signal to noise ratio is low. In this section, we provide the details of our algorithm that uses the above results. The essential steps of our algorithm involve:

- Compute the gradient image, \( \hat{G}(r, c) \), by using a conventional gradient estimation method (Sobel, Canny, etc).

- For each pixel \((r, c)\), compute the integrated gradient magnitudes \( I\hat{G}(r, c, \theta) \) using domains that are directed line intervals \( L'(\theta, K, r, c) \). Note that \( L'(\theta, K, r, c) \) is different from \( L(\theta, K, r, c) \) in that \( L' \) denotes a directed line interval with origin at \((r, c)\) and direction \( \theta \). Also, the domain of \( \theta \) is the interval \([0, 2\pi)\) for \( L' \) whereas the domain of \( \theta \) is the interval \([0, \pi)\) for \( L \). The \( \theta_i \)'s are discrete directions \( \theta_1, \theta_2, \ldots, \theta_D \) specified by the user.

- Let:

  - \( \hat{P}(r, c) = \max_\theta I\hat{G}(r, c, \theta) \)
  - \( \hat{\theta}_{\text{max}}(r, c) = \arg\max_\theta I\hat{G}(r, c, \theta) \)
  - \( \hat{Q}(r, c) = \min_\theta I\hat{G}(r, c, \theta) \)
  - \( \hat{\theta}_{\text{min}}(r, c) = \arg\min_\theta I\hat{G}(r, c, \theta) \)

- Compute \( \hat{\theta}(r, c) \), the likelihood that a pixel \((r, c)\) is an edge pixel, by:

  \[
  \#\{\theta \exists (x, y) \in L'(\theta, K, r, c) \text{ such that } \hat{N}(x, y) < \hat{\theta}(r, c)/D} \]

We make sure that each pixel \((x, y)\) counts only once in the computation since it is possible that a given pixel \((x, y)\) can be part of multiple directed line neighborhoods.

- Threshold the image \( \hat{\theta} \) at 0.5 and determine the connected components image. This image contains the detected features.

- A weighted thinning step is necessary in order to thin the detected result and produce one pixel wide boundaries.

- Sum the squared gradient magnitudes for pixels in each connected component and threshold (specified a given probability of false alarm). Components with summed squared gradient magnitudes above a given threshold are retained.

Notice that the only parameters used in this algorithm is the neighborhood size. The probability threshold of 0.5 results in thick regions that are thinned at a later step. Notice that no absolute threshold on the gradient magnitude was used. A digital image contains boundaries of several types: boundaries due to texture, boundaries due to discontinuities in the objects, occluding boundaries, boundaries due to shadows etc. In order to detect and link all low contrast boundaries, we have dispensed with the estimation of orientation and rather relied on estimates of integrated gradient magnitudes in several directions.

---

1In the current implementation the user specifies \( D \) and the orientation space is discretized into \( D \) angles.
to find the best and worst directions of integrated gradients. The idea used here is that both the minimum integrated gradient as well as the maximum integrated gradient are high at edge pixels. Rather than consider absolute integrated gradients, we have used the relative edge strength (the ratio of the minimum integrated edge strength (this is along the gradient direction of the edge) to the maximum integrated edge strength (this is along the direction tangent to the edge)). Fine changes in the gray levels will be amplified in this step, but the probability thresholding step requires that in order for a pixel to be deemed an edge pixel its relative edge strength should be a maximum over a neighborhood of a sector covering at least 50 percent of the range of possible directions. In addition, the summed squared gradient estimate for pixels along the contours extracted serves as a mechanism for discriminating between significant contours and insignificant ones.

Figure 2 is a subimage of one model board image. Figure 3 is the image of the maximum integrated gradients. Figure 4 is the image of minimum integrated gradients. Notice that the maximum integrated gradient image is blurred where as the minimum integrated image is sharper and shows the dominant locations of edges. Figure 5 is the image of the ratio of the minimum to the maximum integrated gradient. Note how the ratio compensates for shading effects and provides a relative measure of edge strength at each pixel. It should be mentioned here that there have been attempts at using the log gradient image to compensate for shading. We can view the ratio of the minimum to the maximum integrated gradients as a linear combination of the logarithm of minimum integrated gradient image and the logarithm of the maximum integrated gradient image. That is, if \( \hat{G}_{\min}(r,c) \) denotes the minimum integrated gradient image and \( \hat{G}_{\max}(r,c) \) denotes the maximum integrated gradient image, then the logarithm of the ratio:

\[
\log(\hat{G}_{\min}(r,c)/\hat{G}_{\max}(r,c)) = \log(\hat{G}_{\min}(r,c)) - \log(\hat{G}_{\max}(r,c)).
\]

Figure 6 gives the image of 1.0 - \( \hat{G}(r,c) \). Pixels with zero values are the edge pixels. Figure 7 show the inverted averaged edge strength for each connected component derived after thresholding the probability image at 0.5. The dominant edges are shown dark while the less dominant edges are lighter.

We provide a theoretical analysis of the operator and will derive the distribution for the sum of the squared gradient estimates along a given contour in the next section.

3 Theoretical Analysis

We provide a theoretical analysis of the operator in this section. Our edge idealisation in 1D assumes that there is a sequence of ideal gradient values \( G(x, z) = -K \) to \( K \) in a neighborhood centered around the edge pixel. It is assumed that this sequence of values is such that at the edge location \( G(0) \neq G(y) \forall y \neq -K, \ldots, K \). That is the gradient magnitude at the edge pixel is a local maximum within the \( 2K + 1 \) size interval. In 2D, our edge idealisation is that there are sequences of ideal gradient values: \( G(r, c, t, \theta) \) where \( t \) is a discrete index for the parameterized line segment centered at \( r, c \) with orientation \( \theta \) and \( t < |K| \). It is assumed that there are two individual directions \( \theta_M \) and \( \theta_N \), the maximum and minimum gradient directions respectively, such that the ideal summed squared gradient \( S(\theta_M) < S(\theta_{M+1}) < \ldots < S(\theta_N) \) and \( S(\theta_{N+1}) > \ldots > S(\theta_M) \). This assumption essentially states that if we plot the summed squared gradient as a function of angle starting from the minimum direction \( N \), we would see that the function is unimodal and has a peak at index \( N \). This assumption is violated at junctions. We will see that our algorithm needs to be modified to cope with jumps. The perturbation models used are related to our previous work on analysis of edge detection schemes. We have seen in [10] that the ratio of estimated gradient magnitudes \( \hat{G}^2(r,c)/\sigma^2 \) are distributed as non-central chi-square distributions with 2 degrees of freedom and non-centrality parameter \( G(r,c)/\sigma^2 \). Here, \( \sigma^2 \) is the noise variance in the output after the box filtering step. It is clear that due to the fact that the neighborhoods in the operator overlap during estimation, there is dependence among neighboring gradient estimates \( \hat{G}(r,c) \) and \( \hat{G}(r+k,c+j), (k < |K|, j < |K|) \) where \( K \) is the neighborhood window size. Thus, the maximum and minimum integrated gradients are sums of dependent random variables. The distribution of the sums of dependent Gamma distributed random variables have been studied extensively in statistical literature [9]. The chi-square and non-central chi-square distributions are specializations of Gamma distributions. Thus we can use their results directly here.

Rather than discuss the distributions of the minimum and maximum integrated gradients, we discuss how we determine the optimum threshold. The optimum threshold is set by deriving the distribution of the estimated summed squared gradient along a given detected contour. Let \( L \) be the number of pixels in the contour and let \( X_i = \hat{G}^2(i,t), i = 1, \ldots, L \) denote the ratio of the squared gradient estimates to the noise variance at the \( i \)th pixel of the ordered point sequence. \( X_i \) is the scaled squared gradient estimate at the \( i \)th pixel. Let \( Y = \sum_{i=1}^{L} X_i \). This measure is the summed squared gradient along pixels in the detected contour scaled appropriately by the noise variance in the output. When the non-edge idealisation is one where non-edge pixels have true gradient magnitude of zero and gray values are perturbed by i.i.d Gaussian noise we can show that the distribution of the squared gradient estimate is related to the chi-square distribution. The \( X_i \)'s are chi-squared distributed with 2 degrees of freedom. Under this assumption, it can be shown that the probability distribution of \( Y \) is given by:

\[
Prob(Y < y) = \sum_{k=0}^{\infty} \frac{c_k}{2\Gamma\left(\frac{3}{2} + k\right)} \int_0^{y^{1/3}} (u \frac{u}{\theta})^{\frac{1}{2}+k-1} e^{-u/2} du
\]
where: $\lambda_j = \min \lambda_j$, $\lambda_j$'s are the characteristic roots of the $L$ by $L$ matrix of correlation coefficients $\rho_{i,j}$, $i = 1, \ldots, L$, $j = 1, \ldots, L$. These correlation coefficients describe the dependence between squared gradient estimates $X_i$ and $X_j$ along the curve. Under the exponential correlation law, the correlation coefficient estimates is assumed to be $\rho_{k,j} = \rho^{k-j} |_{k,j = 1, \ldots, L}$.

For this special case,

$$\lambda_j = (1 - \rho^2)/(1 - 2\rho \cos \theta_j + \rho^2)$$

$\theta_j$ is the solution of one of the following equations:

$$\sin \left( \frac{L + 1}{2} \theta_j \right) = \rho \sin \left( \frac{L - 1}{2} \theta_j \right)$$
$$\cos \left( \frac{L + 1}{2} \theta_j \right) = \rho \cos \left( \frac{L - 1}{2} \theta_j \right)$$

$c_k$ is determined by the expansion:

$$\prod_{j=1}^{L} \left( \frac{\lambda_j}{\lambda_j^*} \right) \left[ 1 - \left( \frac{1 - \lambda_j^*}{\lambda_j^*} \right) Z \right] = \sum_{k=0}^{\infty} c_k Z^k$$

The threshold that sets the value of the probability $\text{Prob}(Y \leq T) = 1 - \alpha$ is the $T$ corresponding to the probability of false alarm of $\alpha$.

4 Experimental Protocol

The experimental protocol we followed to compare the edge detector with other edge detectors is the same as in [11]. For completeness sake, we give a brief review of the protocol.

Synthetic images of size 51 rows by 51 columns were generated with step edges at various orientations passing through the center pixel $(R, C) = (26, 26)$ in the image. The gray value, $I(r, c)$, at a particular pixel, $(r, c)$, in the synthetic image was obtained by using the function where $\rho = (r - R) \cos(\theta) + (c - C) \sin(\theta)$.

$$I(r, c) = I_{\text{min}}, \quad \rho < 0 \quad (11)$$
$$= I_{\text{max}}, \quad \text{otherwise}.$$ 

$I_{\text{min}}$ and $I_{\text{max}}$ are the gray values in the left and right of the step edge. The variables $R$ and $C$ designate a point in the image on which the step edge boundary lies. In our experiments we set $I_{\text{min}}$ to be 100 and $I_{\text{max}}$ to be 200. We used orientation $\theta$ values of 0, 15, ..., 175 degrees. To generate ramp edges, we averaged images containing the step edges with a kernel of size 4 by 4 so that the resulting ramps have 5 pixels width. To these ramp edge images we added additive Gaussian noise to obtain images with various signal to noise ratios. We define signal to noise ratio as:

$$\text{SNR} = 20 \log \left( \frac{\sigma_x}{\sigma_n} \right) \quad (12)$$

In our initial implementation we assumed independence between the gradient estimates. In this case, $Y$ is a chi-square distributed random variable with $2 \times L$ degrees of freedom.

Figure 1: False Alarm vs Misdetection Characteristics of Integrated Gradient Operator.

where $\sigma_0$ is the standard deviation of the gray values in the input image and $\sigma_n$ is the noise standard deviation. We used SNR values of 0, 5, 10, 20 dB. They correspond to $\sigma_0/\sigma_n$ values of 1, 1.78, 3.162, and 10 respectively. Groundtruth edge images were generated by using the following function where $\rho = (r - R) \cos(\theta) + (c - C) \sin(\theta)$.

$$I_1(r, c) = 0, \quad \rho < -0.5 \quad (13)$$
$$= 1, \quad \text{otherwise}.$$ 

$$I_2(r, c) = 0, \quad \rho < 0.5$$
$$= 1, \quad \text{otherwise}.$$ 

$$I(r, c) = I_1(r, c) \text{ xor } I_2(r, c)$$

We used 5 pixel long directed line intervals for the integrated gradient operator. The edge accuracy evaluation proceeded as follows. The edge pixel location error $E$ is defined as the distance along the gradient direction from the true edge pixel to the nearest labelled edge pixel (if one exists, in the edge detector output). A given ground truth edge pixel is assumed to be missing in the detector output if there are no edge pixels in the detector output within an interval centered on the ground truth edge pixel. The interval is oriented along the gradient direction and the number of pixels in the interval is equal to the edge operator width.

Figure 1 gives the false alarm vs misdetection curve for the new operator. It can be seen that the false alarm rate is as low as 1 percent for misdetection rates of 2 percent when the signal to noise ratio is 3 dB. Moreover the false alarm rate is about 2 percent at the same misdetection rate when the signal to noise ratio is 0dB. More experiments are needed to compare the performance of this operator against current edge detection schemes.
5 Results/Limitations/Extensions

We provide results on real data in this section. Results on an aircraft image obtained from Stanford Univ are shown in Figure 8, 9, and 10. The results were obtained by using a directed interval of 5 pixel width. It can be seen that nearby edges closer than 5 pixels are smoothed. We plan on integrating information from multiple scales to handle this problem. We would process the image by summing the squared gradients over 1, 2, 3, ..., K pixel long directed line intervals. Information obtained from finer scale could be used to decide if processing is allowed to continue at a coarser scale. We are in the process of devising an algorithm that adaptively decides the size of K. It can also be seen that the algorithm results are poor near junctions. A next step would be to modify the algorithm to handle junctions. It can be seen that the preliminary results are encouraging. Very low contrast edges can be detected reliably. This means that effective mechanisms of filtering out clutter information from objects of interest need to be developed in order to facilitate fast recognition. We view this problem as a classification step, one in which the contours produced by our algorithm are classified into several categories: (i.e.) building, non-building, textural edge, etc. We are in the process of developing a classification scheme that uses geometric parameters of the contours as well as the grayscale characteristics along the detected contour.

Since we are dealing with integrated gradients, a fundamental question is the nature of the appropriate gradient estimator to be used in the gradient estimation step. We have not discussed the variety of gradient and contrast estimation schemes that we have tried. We are in the process of developing systematic protocol to characterize the effects of each step in the algorithm. We have also integrated robust estimation techniques for gradient estimation into the algorithm. This has resulted in even better performance. We will have more results to report during the workshop.

6 Conclusion

This paper discussed a new edge detector based on integrated gradients. A theoretical analysis of the operator was done and the algorithm was evaluated by using the experimental protocol developed in [11]. Preliminary results are encouraging and we are in the process of thorough evaluation of the algorithm. We are planning on using this algorithm as one of the steps in a building extraction algorithm sequence. In this paper automatic thresholds were chosen just based on the gradient distributions. However, we plan on integrating geometric parameters as well to classify boundaries into different classes (ex: building/non-building). The sequence we are currently working with is discussed in another paper in this proceedings [13].

Acknowledgments

We would like to thank Sheng-Jyh Wang and Tom Binford for providing us with some of the real images.

References

Figure 2: Original Image

Figure 3: Inverted Maximum Integrated Gradient Image – Dark regions are areas of high integrated gradient.
Figure 4: Inverted Minimum Integrated Gradient Image – Dark areas are areas of high minimum integrated gradient.
Figure 5: Ratio of Minimum to Maximum Integrated Gradient Image

Figure 6: 1.0 - P(Edge), Dark pixels correspond to edge pixels
Figure 7: Edge strength image (before thinning, thresholded at $P(\text{Edge}) = 0.5$)
Figure 8: Aircraft Image: Note that the printer has enhanced the contrast
Figure 9: Results on Aircraft Image (Gradient Estimate - Standard Least Squares Planar Fit)