# Techniques for Assessing Polygonal Approximations of Curves

Paul L. Rosin
Department of Computer Science and Information Systems,
Brunel University,
Middlesex,
UK.
email: Paul.Rosin@brunel.ac.uk

**Abstract**

Given the enormous number of available methods for finding polygonal approximations to curves techniques are required to assess different algorithms. Some of the standard approaches are shown to be unsuitable if the approximations contain varying numbers of lines. Instead, we suggest assessing an algorithm's results relative to an optimal polygon, and describe a measure which combines the relative fidelity and efficiency of a curve segmentation. We use this measure to compare the application of fifteen algorithms to a curve first used by Teh and Chin [31]; their ISEs are assessed relative to the optimal ISE. In addition, using an example of pose estimation, it is shown how goal-directed evaluation can be used to select an appropriate assessment criterion.

# 1 Introduction

Over the last 30 years there has been a substantial and continual interest in the piecewise linear approximation of (mostly plane) curves. In this paper we shall restrict ourself to approaches which require the polygon vertices to lie on the curve. Thus, given a curve $\mathcal{C} = \{x_i, y_i\}_{i=1}^{N}$ the goal is to find the subset of dominant points $\mathcal{D} = \{x_i, y_i\}_{i=1}^{M}$ where $M \leq N$ and $\mathcal{D} \subset \mathcal{C}$. Several algorithms have been described for determining the *optimal* polygonal approximation according to various criteria [6, 17, 19, 30]. Since these algorithms are computationally expensive (usually between $O(N^2)$ and $O(N^3)$) they tend not to be used in practise. Instead many efficient sub-optimal algorithms have been developed, often running in $O(N)$.

Surprisingly, given the plethora of algorithms now available, many authors provide little analysis of their performance, but rely or resort instead to a qualitative demonstration, merely plotting their resulting segmentation. Naturally, this is unsatisfactory since it is difficult to assess the relative merits of the various algorithms, and a more quantitative approach is necessary [10, 13]. Fischler and Wolf [7] rated curve segmentation results using human observers. However, a more convenient and repeatable approach would be preferable. Several recent papers on dominant point detection have quantified their performance based on: the percentage of missed points versus the percentage of false points [14]; the numbers of missed and false points versus different corner angles and different settings of the algorithm's parameters [35]; and location error versus noise standard deviation, corner angle, and curve length [35]. The disadvantage of these approaches is that in order to simplify the problem of requiring ground truth information (i.e. the location of the corners) the algorithms were tested on simple synthetic curves made up from two noise free arcs [14] or two noisy straight lines [35]. Such curves are not necessarily indicative of real curves extracted from images. Kadonaga and Abe [12] used both approaches to compare several algorithms: 1/ invariance under rotation, scaling, and reflection was tested by determining the percentage of similar dominant points detected on the transformed and untransformed curves, and 2/ curve segmentation results were assessed by subjective human evaluation. On individual test cases they found a poor correlation between the two assessment methods, although this was improved by averaging assessments over 10 curves. Further problems encountered with human subjects were the variation in evaluation between subjects, different degrees of confidence in grading different points, and the presence of several possible but mutually exclusive dominant points. A problem when determining numbers of detected or missed dominant points is the need to allow for some shifting of the detected position of the points (e.g. $\pm 1$ pixel). However, the degree of allowable shift should depend on the shape of the curve since a shift is permissible on a low curvature section of curve but not at high curvature sections.

Further considerations were provided by Aoyama and Kawagoe [3] who catalogued the various distortions introduced by the approximation process. In addition to metric displacement and deflection (termed physical distortions) there were also logical distortions. These could arise from the shifting of breakpoints, affecting local geometric features such as corners, spikes, and smooth connecting points (e.g. between a straight and curved section of the curve), as well as parallel and perpendicular lines. At a more global level, qualitative relation distortions include change in topology (e.g. creating self intersections) and the loss of symmetries. Unfortunately, while these are all important issues affecting the performance of polygonal approximation algorithms it is not obvious how to quantify and combine their effects.

# 2 Sarkar's FOM

Most practical interest in assessing polygonisation algorithms has been restricted to quantifying the physical distortions introduced by the approximation process. The two most common measures that are sometimes provided are the compression ratio $CR = \frac{N}{M}$ and the integral square error (ISE) between the curve $\mathcal{C}$ and the approximating polygon. However, there is a tradeoff between these two measures since a high compression ratio leads to an excessive distortion of the polygon (i.e. a high ISE); alternatively, maintaining a low ISE can lead to a low compression ratio. This means that comparing algorithms based on one or the other measure alone is of no value as it does not solve the problem of comparing two or more polygonal approximations with different numbers of lines. To capture this tradeoff Sarkar [29] combined the two measures as a ratio, producing a normalised figure of merit FOM $= \frac{CR}{ISE} = \frac{N}{M \times ISE}$. Similar approaches were used by Held *et al.* [11] and Rosin and West [27].

Unfortunately, Sarkar's FOM is also unsuitable as a measure since it requires the $M \times$ISE term to be constant. Otherwise, approximations with different numbers of lines cannot be compared in a meaningful way. It can be demonstrated that $M \times$ ISE is not constant by analysing the simple example of a circle. The optimal polygonal approximation for most error criteria (including ISE) is a regular polygon inscribed in the circle. For a regular $M$ sided polygon inscribed in a circle of radius $r$ the ISE (i.e. the $E_2$ norm) can be calculated as (see Appendix A)

$$E_2 = Mr^3 \left( \frac{3}{2} \sin \frac{\pi}{M} + \frac{1}{6} \sin \frac{3\pi}{M} - \frac{2\pi}{M} \cos \frac{\pi}{M} \right).$$

Plotting out the theoretical optimal FOM for a circle (figure 1) it can be seen that the measure is biased to favour approximations with large numbers of lines.

There are of course many other criteria available in place of ISE, and we can carry out the same analysis on them to see if any of them would fare better for inclusion in the FOM. For instance, in addition to $E_2$, $E_1$ and $E_\infty$ are popular norms. The $E_1$ error corresponds to the area between the polygon and circle

$$E_1(M) = \pi r^2 - \frac{Mr^2}{2} \sin \frac{2\pi}{M} = r^2 \left( \pi - \frac{M}{2} \sin \frac{2\pi}{M} \right).$$

The $E_\infty$ error uses the maximum deviation between the polygon and circle, given by

$$E_\infty(M) = r \left( 1 - \cos \frac{\pi}{M} \right) = 2r \sin^2 \frac{\pi}{2M}.$$

Lowe [15] suggested that long approximating lines should be permitted greater deviations than short lines, and so he normalised the deviations by dividing them by the length of the approximating line. For a circle this gives

$$E_{\infty/L}(M) = \frac{r\left(1 - \cos \frac{\pi}{M}\right)}{2r \sin \frac{\pi}{M}} = \frac{1}{2} \tan \frac{\pi}{2M}.$$

Another possibility, used by Rosin and West [27], is to sum the normalised maximum deviations from each line segment, giving

$$E_{\Sigma\infty/L}(M) = \frac{1}{2M} \tan \frac{\pi}{2M}.$$

Rather than normalise the maximum deviation Wall and Danielson [34] normalised the area deviation

$$E_{1/L}(M) = \frac{r\left(\pi - \frac{M}{2} \sin \frac{2\pi}{M}\right)}{2 \sin \frac{\pi}{M}}.$$

A rather different measure, which was maximised by Sato [30], is the length of the polygon

$$L = 2Mr \sin \frac{\pi}{M}.$$

However, it can be seen that none of the above criteria provide a FOM that is constant for varying values of $M$. Of course, using the circle model the FOM could be corrected to provide invariance in $M$. For instance, using $E_{\Sigma\infty/L}$ the corrected FOM would be

$$FOM = \frac{CR}{E_{\Sigma\infty/L}} \tan \frac{\pi}{2M}.$$

Alternatively, if the summed absolute difference in orientation (SADO) between the sides of the polygon and the corresponding sections of the curve is used, then for a circle the error is

$$E_\theta(M) = \frac{2\pi^2 r}{M}$$

which *would* be suitable for the FOM since it is a linear function of $M$, and so $\frac{CR}{E_\theta}$ is constant over all $M$. However, for more general curves there is another problem relating to the effect that natural scales have on the error of the polygonal approximation.

The natural scales of a curve are those scales at which the curve displays some qualitative shape which is distinctive compared to other scales [26]. Previously we detected these scales by performing Gaussian smoothing. For instance, consider a circle superimposed with a sine wave as shown in figure 2a. Smoothing the curve over a large range of scales will have little effect except for distorting the bumps, and so all these smoothed versions of the curve are qualitatively similar. Eventually increasing the smoothing will eliminate the bumps, resulting in a simple circle. Thus the curve has two natural scales: the bumpy circle, and the simple circle. The errors of an approximation polygon will vary considerably according to which natural scale the approximation is taking place at. For instance, for the sinusoidal circle, when minimising ISE, the first series of polygons ($3 \rightarrow 7$ lines) are regular polygons which are just approximating the circular component. The next set of polygons ($8 \rightarrow 28$ lines) coarsely approximate the bumps by triangles. Finally, increasing the number of sides improves the approximation to the sinusoidal shape of the bumps. This is experimentally verified by finding the optimal polygons with respect to some of the above criteria. For instance, looking at the plots of ISE and SADO against number of sides in figures 2b and 2c the three qualitatively distinct sets of polygons over the above ranges are evident as three sections of the curve with different slopes. Some of the polygons obtained when minimising ISE are plotted in figure 3. The regular polygons at $M = 3$ and 7 are shown. After $M = 7$ the additional sides introduce bumps which are added one at a time (e.g. $M = 9$ and 19) until all the bumps are represented ($M = 28$). The rate of reduction of error reduces thereafter. Even with $M = 70$ sides the increased definition of the sinusoidal undulations is relatively minor.

# 3 Relative Measures

The qualitatively different structures of a curve's shape at different scales means that it would be extremely difficult to find an effective figure of merit that can be fairly applied to compare approximations with different numbers of points that is only a function of an absolute error measure taken at a single scale (i.e. a fixed value of $M$). Another approach, taken by Ventura and Chen [33], was to assess their algorithms with respect to the reference segmentation of an optimal algorithm. They used the percentage relative difference, calculated as

$$PRD = \frac{E_{approx} - E_{opt}}{E_{opt}} \times 100$$

where $E_{approx}$ is the error incurred by the suboptimal algorithm to be tested, and $E_{opt}$ is the error incurred by the optimal algorithm; both algorithms are set to produce the same number of lines. This approach has the significant advantage that it enables approximations with any number of lines to be compared. We advocate a similar method, but will first split the assessment into two components: fidelity and efficiency. Fidelity measures how well the suboptimal polygon fits the curve relative to the optimal polygon in terms of the approximation error. Efficiency measures how compact the suboptimal polygonal representation of the curve is, relative to the optimal polygon which incurs the same error. They are defined as

$$\text{Fidelity} = \frac{E_{opt}}{E_{approx}} \times 100$$

$$\text{Efficiency} = \frac{M_{opt}}{M_{approx}} \times 100$$

where $M_{approx}$ is the number of lines in the approximating polygon produced by the suboptimal algorithm and $M_{opt}$ is the number of lines that the optimal algorithm would require to produce the same error as the suboptimal algorithm (i.e. $E_{approx}$). Since an exact value of $M_{opt}$ is not generally available it is calculated by linear interpolation of the two closest integer values of $M$ produced by the optimal algorithm.

Depending on the shape of the curve, the two measures may vary considerably. A combined measure is taken as the root of their product

$$\text{Merit} = \sqrt{\text{Fidelity} \times \text{Efficiency}} = \sqrt{\frac{E_{opt} M_{opt}}{E_{approx} M_{approx}}} \times 100.$$

One problem remains – how should the error be quantified? In section 2 we listed various possible measures: $E_1$, $E_2$, $E_\infty$, $E_{\infty/L}$, $E_{\Sigma\infty/L}$, $L$, and $E_\theta$, any one of which may be suitable. However, they were all based on the approximating polygon. Since dominant point detection algorithms do not explicitly assume a connecting polygon then some other type of error measure may be more appropriate. For instance, the approximation criteria of local symmetry [18] or stability of curvature maxima over scale [21] used to detect dominant points may be applicable. One approach to tackle the suitability of different measures is to apply goal-directed evaluation [32]. Given that the curves are being approximated for some specific task (e.g. construction of higher level features or model matching) the appropriateness of various error measures could be evaluated with respect to the benefits they provide the task.

# 4 An Example: Teh and Chin's Curve

To demonstrate the new measures we will apply them to the results of various algorithms which have been applied to the curve presented in Teh and Chin [31] (see figure 4). Teh and Chin tabulated the ISE of many algorithms when applied to this curve, and the results of applying other algorithms have been subsequently provided in the literature (either visually or with their ISE). Therefore we shall analyse the algorithms with reference to the optimal polygonal representation using the $E_2$ criterion. The optimal solutions were found using dynamic programming (e.g. Perez and Vidal [19]) run for all values of $M = [3, 50]$. In table 6 are shown the results of the algorithms with the parameters as specified by Teh and Chin or the original authors as applicable. Melen and Ozanian's [16] algorithm was run with $s = 4$ and $t = 10$. Since Lowe's algorithm makes no provision for closed curves the starting point was selected by hand. The modified version of Lowe's algorithm by Rosin and West [27] which includes a merging stage was also applied, but the results for this example were identical to Lowe's. Figure 5 plots the curve of the optimal ISE error for all the values of $M$ as well as the results of the various suboptimal algorithms.

We can see that for this example Lowe's algorithm performs extremely well – close to the optimum, achieving a merit rating of 97.1, substantially outperforming all the other algorithms. For instance, Melen and Ozanian's algorithm incurred five times the ISE using the same number of lines, thereby receiving a merit rating of only 28.8, ranking 17th. More valuable is the ability to compare the results of very different polygonal approximations with different numbers of lines. For instance, although Rosenfeld and Weszka (a), Teh and Chin, and Rosenfeld and Johnston (b) respectively generated 14, 22, and 30 points with associated ISEs of 59.12, 20.61, and 8.85, it is now possible to see that they are all roughly on par since they have the similar ratings of 43.8, 44.9, and 44.7. Figure 6 shows the resulting polygons produced by Lowe's and by Sankar and Sharma's algorithms, which respectively received the highest and lowest ranking.

Of course, to properly assess the general effectiveness of the algorithms they would need to be tested on many more curves. Moreover, since most of the algorithms have some sort of scale parameter, these algorithms should be tested over the full range of scales. For instance, figure 7 shows the ISE curves plotted for the algorithms of Melen and Ozanian ($s = 2$ and varying $t$), Ramer, and Rosenfeld and Johnston. Unlike the optimal algorithm they do not display ISE that monotonically decreases with increasing numbers of points. Where Rosenfeld and Johnston's algorithm produced several different ISE values for the same number of points the lower value was plotted. This highlights the problem that assessing an algorithm at a single scale may not accurately reflect its performance at different values of its scale parameter. This can be solved by averaging the merit values over all scales, giving 26.1 for Melen and Ozanian, 71.8 for Ramer, and 44.5 for Rosenfeld and Johnston. Another problem is that the relationship between an algorithm's parameter may not predictable reflect the scale of analysis. This is demonstrated by plotting the number of points detected by Rosenfeld and Johnston's algorithm against the smoothing parameter $m$ in figure 8 – several fluctuations are evident.

# 5 Goal-Directed Evaluation

We suggested in section 3 that one method of determining the most suitable error metric for polygonisation would be to perform goal-directed evaluation. In this section this is demonstrated with an example in which the task is to perform pose estimation. The polygonal approximations (using a fixed number of lines) are found for both a model and a test curve. Assuming that the algorithms provide stable results with respect to perturbations of the curve there should be no correspondence problem between points on the curves. The pose of the test curve is then found by performing the 2D transformation between the corresponding points of the model and test curve polygons that minimises the least square error. The metrics are assessed on the correctness of the estimated pose.

The tests are based on the synthetic curve used by Rosin and West [27] which has been downscaled by a factor of 5 and subsampled to speed up testing. The uncorrupted curve is shown in figure 9a and contains 449 pixels. For each approximation criterion to be tested the optimal segmentation of the curve is determined using dynamic programming. For instance, figure 9b shows the 25 line polygon found using the $E_\infty$ metric. The curve is corrupted by adding noise at two levels of scale by the following process:

- the curve is subsampled at every fourth point,
- these points are perturbed by adding Gaussian noise ($\sigma = 1$)
- the curve is reconstructed using cubic splines, and
- further Gaussian noise ($\sigma = 0.1$) is added to this dense set of points.

The resulting curve has around 500 pixels. Two examples of noisy curves are shown in figure 9c and 9d.

Six hundred noisy curves were generated. They were not transformed, and so the expected rotation and $X$ and $Y$ translation is zero. All models and test curves were segmented into polygons containing 25 lines. This number was just more than sufficient to capture all the features on the model curve. The approximation criteria test were: $E_1$, $E_2$, $E_\infty$, which were also normalised by the length of the lines: $E_{1/L}$, $E_{2/L}$, $E_{\infty/L}$, and $L$, and $E_\theta$. Another criterion considered was the product of the SADO and the maximum deviation for each approximating line, $E_{\theta\infty}$. The motivation was to combine the favourable properties of each (Aoyama and Kawagoe [3] list the advantages and disadvantages of the displacement and deflection approximation criteria).

Table 6 shows the RMS errors of the model transformation parameters for the different polygonalisation criteria. It can be concluded that for this pose estimation problem the $E_1$, $E_2$, and $E_\infty$ criteria appear most suitable as they provide lower RMS errors than the other criteria.

# 6 Conclusions

This paper tackles the need for a technique that is able to assess different algorithms for finding polygonal approximations to curves. The standard approaches which use ISE, CR, or their combination as $\frac{CR}{ISE}$ are shown to be unsuitable if the approximations contain varying numbers of lines. Moreover, it was shown that if a curve contains several natural scales (which is the case for all non-trivial curves) then no measure that does not take the shape of the curve into account is likely to be suitable. In their place it is suggested that to ensure invariance to the numbers of lines an algorithm's results should be assessed relative to some "gold standard" which must be available for each possible number of lines. The proposed assessment combines the fidelity and efficiency of an algorithm's results, i.e. how well the suboptimal polygon fits the curve, and the compactness of the suboptimal polygonal representation of the curve.

Of course, this leaves the problem of what to choose as the gold standard. Our solution is to use the optimal polygon for the given number of lines according to some criterion. An example was given using the $E_2$ measure to test fifteen algorithms applied to Teh and Chin's curve. More generally, the appropriate criterion can be selected according to the task that the lines are intended for. An example of performing such a goal-directed evaluation was given for a pose estimation task, in which case the $E_1$, $E_2$, and $E_\infty$ criteria appeared most suitable. Thus it suggests that the use of ISE in the test on Teh and Chin's curve was not critical, and that an evaluation using $E_1$ or $E_\infty$ would give similar results. The goal-directed approach could be applied to rate the polygonisation algorithms based on their performance on the task. However, it is often probably more useful to have a task-independent assessment of the algorithms so that they do not have to be re-evaluated for every individual task.

A weakness of the assessment criteria used in this paper is that they do not directly evaluate the various distortions described by Aoyama and Kawagoe [3], particularly since the specific aim of some algorithms [3, 9] is represent significant points (e.g. spikes) extremely accurately, potentially at the cost of increasing other errors. Also not considered in this paper is the assessment of an algorithm's robustness under systematic distortions of the data, e.g. blurring, ranges and different types of noise, rotations, scaling (including subsampling), and occlusion (i.e. deletion of the ends of open curves). The latter task should prove straightforward, but the former problem of determining and then measuring such structural deviations is more difficult, and is an open area for investigation.

# Appendix A – The optimal ISE of a polygonal approximation to a circle

From the diagram in figure 10 it can be seen that the ISE of the approximation of a circle by a regular $M$ sided polygon can be calculated as $M$ times the integral of the squared distance between the circle and the top horizontal side of the polygon. This is given by

$$\begin{aligned}
E_2 &= 2M \int_0^{r\cos\left(\frac{\pi}{2}-\frac{\pi}{M}\right)} \left[\sqrt{r^2-x^2} - r\sin\left(\frac{\pi}{2}-\frac{\pi}{M}\right)\right]^2 dx \\
&= 2M \int_0^{r\sin\frac{\pi}{M}} \left[\sqrt{r^2-x^2} - r\cos\frac{\pi}{M}\right]^2 dx \\
&= 2M \int_0^{r\sin\frac{\pi}{M}} r^2\left(1+\cos^2\frac{\pi}{M}\right) - x^2 - 2r\cos\frac{\pi}{M}\sqrt{r^2-x^2}\,dx \\
&= 2M \left\{ \left[ xr^2\left(1+\cos^2\frac{\pi}{M}\right) - \frac{x^3}{3}\right]_0^{r\sin\frac{\pi}{M}} - 2r\cos\frac{\pi}{M}\left[\frac{x}{2}\sqrt{r^2-x^2} + \frac{r^2}{2}\sin^{-1}\frac{x}{r}\right]_0^{r\sin\frac{\pi}{M}} \right\}
\end{aligned}$$

which after some simplification reduces to

$$E_2 = Mr^3\left(\frac{3}{2}\sin\frac{\pi}{M} + \frac{1}{6}\sin\frac{3\pi}{M} - \frac{2\pi}{M}\cos\frac{\pi}{M}\right).$$

# References


[1] I.M. Anderson and J.C. Bezdek. Curvature and tangential deflection of discrete arcs. *IEEE Trans. PAMI*, 6:27–40, 1984.

[2] N. Ansari and K.W. Huang. Non-parametric dominant point detection. *Pattern Recognition*, 24:849–862, 1991.

[3] H. Aoyama and M. Kawagoe. A piecewise linear approximation method preserving visual feature points of original figures. *CVGIP: Graphical Models and Image Processing*, 53:435–446, 1991.

[4] C. Arcelli and G. Ramella. Finding contour-based abstractions of planar patterns. *Pattern Recognition*, 26:1563–1577, 1993.

[5] P.C. Chung, C.T. Tsai, and Y.N. Sun. Polygonal approximation using a competitive hopfield neural network. *Pattern Recognition*, 27:1505–1512, 1994.

[6] J.G. Dunham. Optimum uniform piecewise linear approximation of planar curves. *IEEE Trans. PAMI*, 8:67–75, 1986.

[7] M.A. Fischler and H.C. Wolf. Locating perceptually salient points on planar curves. *IEEE Trans. PAMI*, 16:113–129, 1994.

[8] H. Freeman and L.S. Davis. A corner-finding algorithm for chain-coded curves. *IEEE Trans. Comp.*, 26:297–303, 1977.

[9] F. Gritzali and G. Papakonstantinou. A fast peicewise linear approximation algorithm. *Signal Processing*, 5:221–227, 1983.

[10] R.M. Haralick. Performance characterization in image analysis: Thinning, a case in point. *Pattern Recognition Letters*, 13:5–12, 1992.

[11] A. Held, K. Abe, and C. Arcelli. Towards a hierarchical contour description via dominant point detection. *IEEE Trans. on Sys., Man, and Cybernetics*, 24:942–949, 1994.

[12] T. Kadonaga and K. Abe. Comparison of methods for detecting corner points from digital curves. In *Int. Workshop on Graphics Recognition*, 1995.

[13] T. Kanungo, M.Y. Jaisimha, J. Palmer, and R.M. Haralick. A methodology for quantitative performance evaluation of detection algorithms. *IEEE Trans. Image Processing*, 4:1667–1673, 1995.

[14] J. Koplowitz and S. Plante. Corner detection for chain coded curves. *Pattern Recognition*, 28:843–852, 1995.

[15] D.G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31:355–395, 1987.

[16] T. Melen and T. Ozanian. A fast algorithm for dominant point detection on chain-coded contours. In *Proc. 5th Int. Conf. Computer Analysis of Images and Patterns*, 1993.

[17] U. Montanari. A note on minimal length polygonal approximation to a digitized curve. *Comm. ACM*, 13:41–47, 1970.

[18] H. Ogawa. Corner detection on digital curves based on local symmetry of the shape. *Pattern Recognition*, 22:351–357, 1989.

[19] J.C. Perez and E. Vidal. Optimum polygonal approximation of digitized curves. *Pattern Recognition Letters*, 15:743–750, 1994.

[20] U. Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer, Graphics and Image Processing*, 1:244–256, 1972.

[21] A. Rattarangsi and R.T. Chin. Scale-based detection of corners of planar curves. *IEEE Trans. PAMI*, 14:430–449, 1992.

[22] B.K. Ray and K.S. Ray. An algorithm for detection of dominant points and polygonal approximation of digitized curves. *Pattern Recognition Letters*, 12:849–856, 1992.

[23] B.K. Ray and K.S. Ray. Detection of significant points and polygonal approximation of digitized curves. *Pattern Recognition Letters*, 12:443–452, 1992.

[24] A. Rosenfeld and E. Johnston. Angle detection on digital curves. *IEEE Trans. Comp.*, 22:875–878, 1973.

[25] A. Rosenfeld and J.S. Weszka. An improved method of angle detection on digital curves. *IEEE Trans. Comp.*, 24:940–941, 1975.

[26] P.L. Rosin. Representing curves at their natural scales. *Pattern Recognition*, 25:1315–1325, 1992.

[27] P.L. Rosin and G.A.W. West. Non-parametric segmentation of curves into various representations. *IEEE Trans. PAMI*, 17:1140–1153, 1995.

[28] P.V. Sankar and C.V. Sharma. A parallel procedure for the detection of dominant points on a digital curve. *Computer, Graphics and Image Processing*, 7:403–412, 1978.

[29] D. Sarkar. A simple algorithm for detection of significant vertices for polygonal approximation of chain-coded curves. *Pattern Recognition Letters*, 14:959–964, 1993.

[30] Y. Sato. Piecewise linear approximation of plane curves by perimeter optimization. *Pattern Recognition*, 25:1535–1543, 1992.

[31] C.H. Teh and R.T. Chin. On the detection of dominant points in digital curves. *IEEE Trans. PAMI*, 11:859–872, 1989.

[32] Ø.D. Trier and A.K. Jain. Goal-directed evaluation of binarization methods. *IEEE Trans. PAMI*, 17:1191–1201, 1995.

[33] J.A. Ventura and J.M. Chen. Segmentation of two-dimensional curve contours. *Pattern Recognition*, 25:1129–1140, 1992.

[34] K. Wall and P.E. Danielsson. A fast sequential method for polygonal approximation of digitized curves. *Computer Vision, Graphics and Image Processing*, 28:220–227, 1984.

[35] X. Zhang, R.M. Haralick, and V. Ramesh. Corner detection using the map technique. In *Proc. 12th Int. Conf. Pattern Recognition*, pages 549–551, 1994.

| METHOD | #POINTS | ISE | FIDELITY | EFFICIENCY | MERIT | RANK |
|---|---|---|---|---|---|---|
| Rosenfeld & Johnston [24] (a) | 12 | 92.37 | 28.1 | 58.7 | 40.6 | 15 |
| Rosenfeld & Johnston (b) | 30 | 8.85 | 29.8 | 67.2 | 44.7 | 12 |
| Rosenfeld & Weszka [25] (a) | 14 | 59.12 | 29.4 | 65.3 | 43.8 | 13 |
| Rosenfeld & Weszka (b) | 34 | 15.40 | 12.5 | 43.1 | 23.2 | 20 |
| Freeman & Davis [8] (a) | 17 | 79.53 | 15.4 | 45.7 | 26.5 | 19 |
| Freeman & Davis (b) | 19 | 23.31 | 43.1 | 65.8 | 53.3 | 9 |
| Sankar & Sharma [28] | 10 | 769.53 | 5.1 | 41.5 | 14.5 | 20 |
| Anderson & Bezdek [1] (a) | 18 | 36.14 | 31.0 | 57.9 | 42.4 | 14 |
| Anderson & Bezdek (b) | 29 | 6.43 | 46.7 | 78.2 | 60.4 | 6 |
| Teh & Chin [31] | 22 | 20.61 | 34.0 | 59.2 | 44.9 | 11 |
| Ansari & Huang [2] | 28 | 17.83 | 18.8 | 49.5 | 30.5 | 17 |
| Melen & Ozanian [16] | 13 | 122.44 | 16.9 | 49.1 | 28.8 | 18 |
| Sarkar [29] 1 point method | 19 | 17.38 | 57.8 | 73.7 | 65.3 | 4 |
| Sarkar 2 point method | 20 | 13.65 | 66.0 | 78.9 | 72.2 | 3 |
| Lowe [15] | 13 | 21.66 | 95.7 | 98.6 | 97.1 | 1 |
| Ray & Ray [23] (1) | 29 | 11.82 | 25.4 | 60.0 | 39.0 | 16 |
| Ray & Ray [22] (2) | 27 | 11.5 | 32.2 | 65.6 | 46.0 | 10 |
| Arcelli & Ramella [4] | 10 | 75.10 | 51.8 | 80.3 | 64.5 | 5 |
| Held, Abe & Arcelli [11] | 17 | 28.50 | 42.9 | 68.3 | 54.1 | 8 |
| Rattarangsi & Chin [21] | 9 | 130.13 | 48.1 | 69.1 | 57.7 | 7 |
| Ramer [20] | 26 | 5.27 | 76.9 | 92.6 | 84.4 | 2 |
| Chun et al [5] (a) | 14 | 45.60 | 38.1 | 69.4 | 51.4 | x |
| Chun et al [5] (b) | 22 | 12.36 | 56.7 | 76.7 | 66.0 | x |
| Chun et al [5] (c) | 28 | 9.69 | 34.6 | 69.1 | 48.9 | x |

Table 1: Assessment of various algorithms applied to the curve in figure 4

| CRITERIA | A | X | Y |
|---|---|---|---|
| $E_1$ | 10.36 | 8.91 | 9.91 |
| $E_2$ | 9.39 | 8.50 | 8.92 |
| $E_\infty$ | 9.87 | 8.40 | 9.72 |
| $E_{1/L}$ | 36.26 | 14.37 | 37.93 |
| $E_{2/L}$ | 14.36 | 14.82 | 10.19 |
| $E_{\infty/L}$ | 29.65 | 13.69 | 30.27 |
| $L$ | 21.21 | 24.56 | 11.04 |
| $E_\theta$ | 18.54 | 14.25 | 21.83 |
| $E_{\theta\infty}$ | 34.16 | 11.27 | 35.62 |

Table 2: Assessment of various criteria for pose estimation by RMS errors

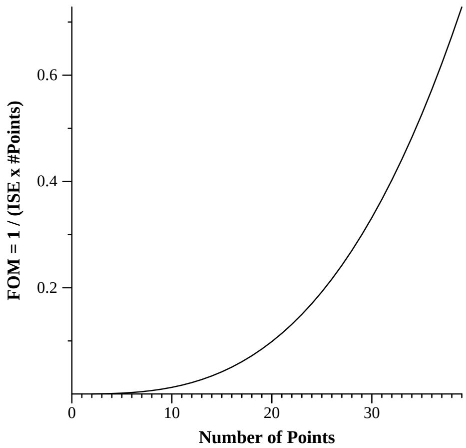
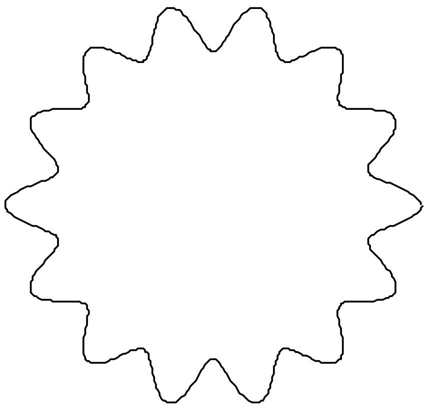
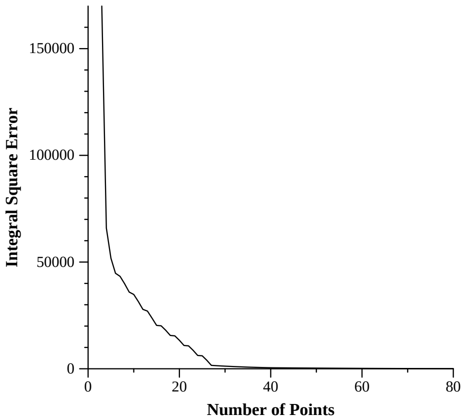


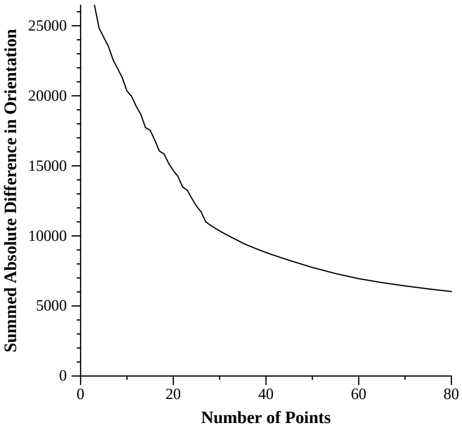

Figure 1: Theoretical FOM of optimal polygonal approximation for a circle ($r = 10$)



(a) Circle with superimposed sine wave

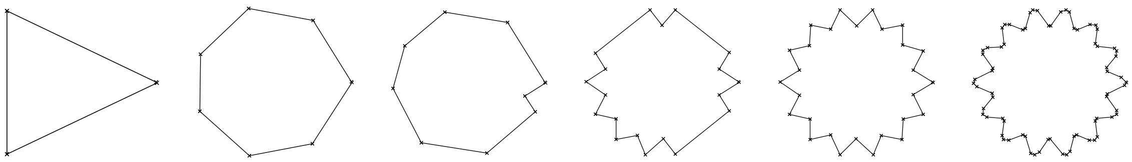



(b) ISE of optimal polygonal approximation




(c) SADO of optimal polygonal approximation

Figure 2: Effects of natural scales on approximation error



$M = 3$ $M = 7$ $M = 9$ $M = 19$ $M = 28$ $M = 70$

Figure 3: Effects of natural scales on the shape of the approximations

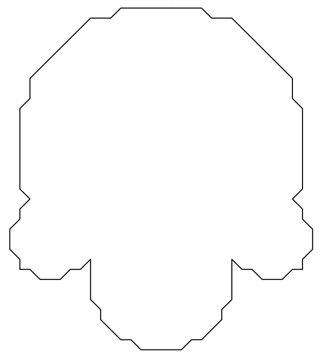


Figure 4: Curve used by Teh and Chin

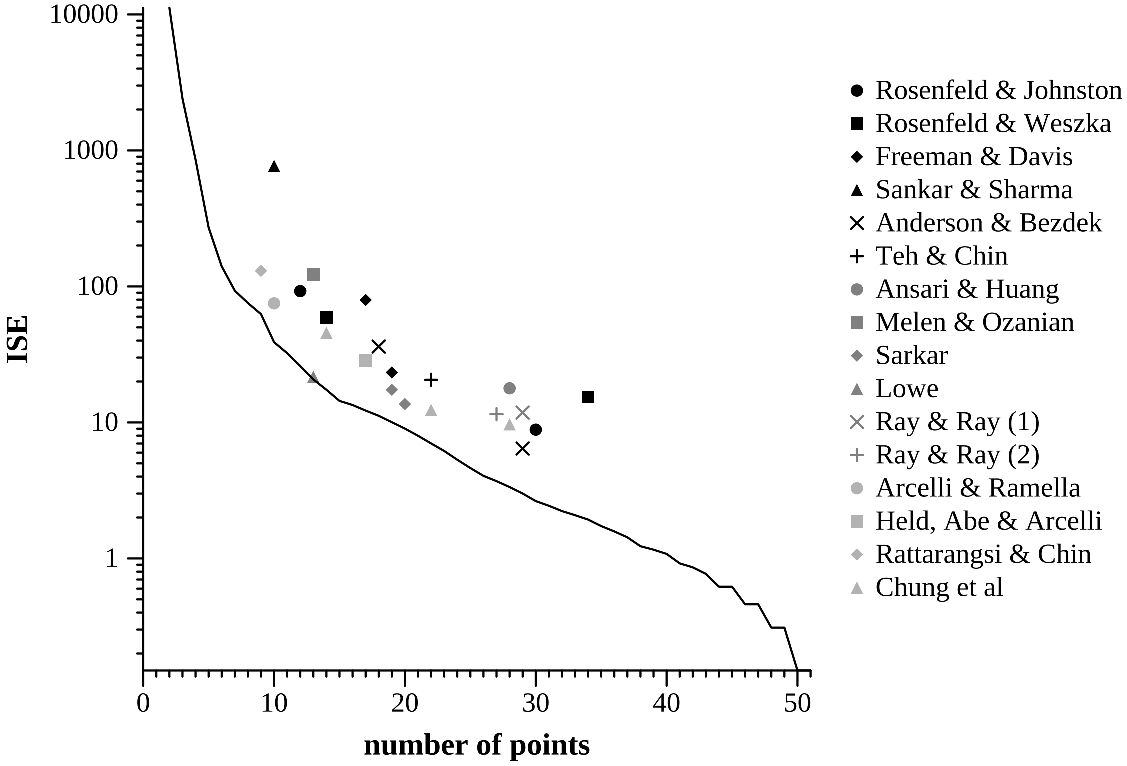



Figure 5: ISE of optimal and other algorithms

(a) Result of Lowe's algorithm

(b) Result of Sankar & Sharma's algorithm

Figure 6: Some approximations of figure 4

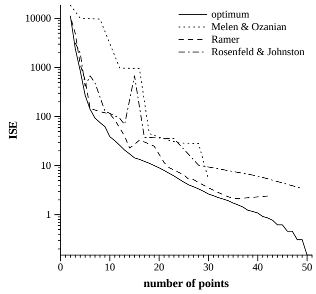



Figure 7: ISE curves of optimal and other algorithms

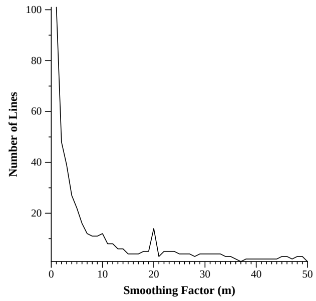
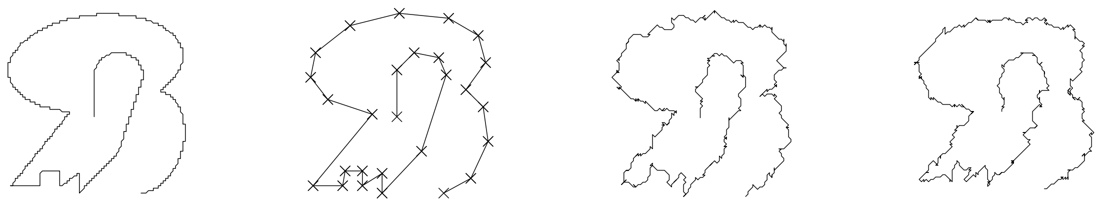



Figure 8: Effect of smoothing parameter on the detected number of dominant points in Rosenfeld and Johnston's algorithm



(a) Uncorrupted test curve (b) Model segmentation using $E_\infty$ metric (c) Test curve with noise (d) Test curve with noise

Figure 9: Examples of curves used for evaluating metrics

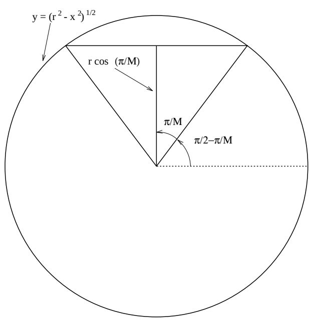



Figure 10: The geometry of a circle with its inscribed polygon