SHAPE ANALYSIS ALGORITHM BASED ON INFORMATION THEORY

D. L. Page, A. F. Koschan, S. R. Sukumar, B. Roui-Abidi, M. A. Abidi

Imaging, Robotics, and Intelligent Systems Laboratory
University of Tennessee, Knoxville, Tennessee 37996-2100, davidpage@ieee.org

ABSTRACT

In this paper, we describe an algorithm to measure the shape complexity for discrete approximations of planar curves in 2D images and manifold surfaces for 3D triangle meshes. We base our algorithm on shape curvature, and thus we compute *shape information* as the entropy of curvature. We present definitions to estimate curvature for both discrete curves and surfaces and then formulate our theory of shape information from these definitions. We demonstrate our algorithm with experimental results.

1. INTRODUCTION

Claude Shannon's seminal paper in 1948 [1] launched a revolutionary field of mathematics that has become known as *information theory*. Shannon's work formulates a powerful and general theory to quantize information and in particular to describe information in a probabilistic framework. Researchers have applied information theory to such diverse topics as communications, astronomy, crystallography, and nuclear physics. Our goal in this paper is to apply this theory to geometric contours and shapes. We present an algorithm to compute the *shape information* for 2D planar contours and 3D triangle meshes.

Our motivation for such a metric results from an interest to measure the complexity of a shape—whether a contour or surface—in a wide range of applications from computer vision [2], medical imaging [3], and spatial databases [4]. Consider the sphere and bore pin in Fig. 1. Intuitively, the pin appears more complex than the sphere. How can we quantize our qualitative intuition? In this paper, we propose a computer algorithm that answers this question. We propose that the sphere contains zero ($H_\triangle = 0$ bits) shape information while the pin contains some non-zero value ($H_\triangle = 7.5$ bits). The contributions of this paper are:

- definition of shape information for manifold surfaces and
- development of an algorithm to compute shape information for triangle meshes.

1.1. Previous Work

We find interest in shape complexity in a diverse array of applications from satellite imagery [2] to neuron morphology [3]. King and Rossignac [5] present an interesting paper that is directly relevant to triangle mesh data sets. In this work, the authors consider lossy mesh compression and focus on vertex reduction and bits per vertex minimization. Toussaint [6] proposes another measure for shape complexity in 2D based on polygon decomposition. Chazelle and Incerpi [7] proposed the sinuosity as a measure of complexity where sinuosity is the number of times that a polygon's boundary alternates between complete spirals of opposite orientations [6]. An interesting paper [3] develops a multiscale approach to measure shape. Bending energy is an interesting measure of shape where [8] characterize the contours of biological objects. Vliet and Verbeek [9] extend bending energy definitions to 3D data sets. The key methods that have inspired our approach are the ones that employ information theory to shape description. Oddo [2] developed a segmentation algorithm based on global shape entropy to extract building boundaries from aerial imagery. The entropy definitions in Oddo follow from the gray level definitions of entropy in [10].

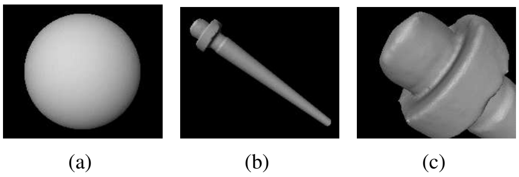


Fig. 1. Which 3D object contains more shape information: (a) the sphere or (b) the bore pin? (c) Zoom view of pin head.

1.2. Information Theory

Recall that from information theory [1] we can define a random variable x with a probability density function (pdf) $p(x)$ that describes the statistics of x. We define the *entropy* H with the following equation:

$$H(x) = -\int_{-\infty}^{\infty} p(x) \log p(x) dx \ , \tag{1}$$

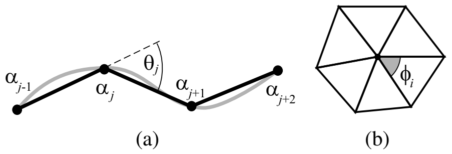
where we assume that $p(x)$ is a continuous function. If the log function is base two, then H is in units of bits. Suppose however that we discretize $p(x)$ such that $p_i = \int_{x_{i-1}}^{x_i} p(x)dx$ where x_i and x_{i+1} are specific values of x. We can now write a discrete formulation of Eq. (1) as

$$H(x) = -\sum_i p_i \log p_i \ . \tag{2}$$

2. PLANAR 2D CURVES

First, we consider the continuous case for 2D curves. Using [11], we arbitrarily define a planar curve $\alpha : I \rightarrow \mathbf{R}^2$ parameterized by arc length s such that we have $\alpha(s)$. We carefully choose, without loss of generality, this parameterization such that the vector field $T = \alpha'$ has unit length. With this construction, the derivative $T' = \alpha''$ measures the way the curve is turning in $\mathbf{R}^2$ and

0-7803-7750-8/03/$17.00 ©2003 IEEE.



(a) (b)

Fig. 2. Curvature estimation for discrete curves and surfaces.

we term T' the curvature vector field. Since T' is always orthogonal to T, i.e. *normal* to α, we can write $T' = \kappa N$ where N is the normal vector field. The real-valued function κ where $\kappa(s) = \|\alpha''(s)\|$ is the curvature function of α and completely describes the shape of α in $\mathbf{R}^2$, up to a translation and rotation. This curvature function is what we intend to exploit to define the shape information of a curve.

We now recast the curvature function $\kappa(s)$ for a family of curves as a random variable. An ensemble of curves with similar statistics over their associated curvature functions have some equivalence in a random variable context. Consider a straight line and a half circle, which both have constant curvature $\kappa(s) = 0$ and $\kappa(s) = \frac{1}{r}$, respectively. The subsequent statistics of these functions are similar in that the variance of each is zero—a direct consequence of a constant function—and thus their pdfs have an equivalent shape with only an offset in the translation of their means. We define these pdfs with the impulse function $\delta(x)$ such that $\delta(0) \to \infty$ and $\int_{-\infty}^{\infty} \delta(x)dx = 1$. We thus have $p(k) = \delta(k)$ for the straight line and $p(k) = \delta(k - \frac{1}{r})$ for the half circle where k is the random variable for the $\kappa(s)$ signal. If we apply Eq. (1), both curves contain no information. Intuitively, this result makes sense because both curves exhibit no variation in shape and thus possess zero *shape information*.

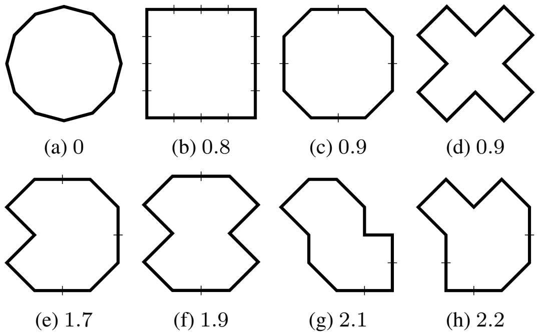
Next, we consider the discrete case. We formulate shape information from discrete samples of continuous curves. For planar curve α, we have samples $\alpha_j = \alpha(s_j)$. We assume that we sample uniformly across the arc length of the curve such that $\Delta s = s_j - s_{j-1}$ is a constant. This approach leads to N samples over the curve of α. Since we have uniform sampling along the curve, the curvature κ_j is directly proportional to the turning angle θ_j formed by the line segments from endpoint α_{j-1} to endpoint α_j and from α_j to α_{j+1}. We illustrate these concepts in Fig. 2(a).

We now need to estimate the pdf of the curvature function from the θ_j estimates. To do so, we choose a number of bins M and associate a probability with each bin. We assume a uniform bin width $\Delta\theta$ such that $\Delta\theta = \frac{\theta_{max} - \theta_{min}}{M}$. The bin probability p_i becomes

$$p_i = \frac{B_i}{N} \ , \tag{3}$$

where B_i is the number of θ_j samples that fall in the range $\theta_{min} + (i-1)\Delta\theta \leq \theta_j < \theta_{min} + i\Delta\theta$. In practice, we choose a bin width $\Delta\theta$, which then determines the number of bins M. If we fix $\Delta\theta$, we can compare entropy computations among various shapes.

With these definitions, we can compute the discrete shape information for a variety of uniformly sampled curves. Consider the discrete closed curves in Fig. 3. The sample points α_j are the joints of line segments and hash marks for adjoining flat segments. For example, Fig. 3(b) has sixteen samples with four samples at the corners and twelve on the flat edges. The corner samples have a $\frac{\pi}{2}$ turning angle while the flat samples have a zero angle. Thus, the bin probabilities are $p_0 = \frac{3}{4}$ and $p_{\frac{\pi}{2}} = \frac{1}{4}$. This formulation leads to the entropy $H = 0.8$ bits.



(a) 0 (b) 0.8 (c) 0.9 (d) 0.9

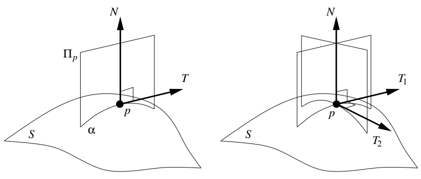
(e) 1.7 (f) 1.9 (g) 2.1 (h) 2.2

Fig. 3. These line-segment curves are discrete approximations of smooth curves. Below each contour we show the shape information in bits.

We should add a few comments. First, note the regular polygon in Fig. 3(a). Since we assume uniform sampling, the original surface is a constant curvature circle. As noted previously, such a curve has zero information, and as expected, our discrete formulation also gives rise to zero information. So, our discrete formulation fits nicely with the continuous case. Second, with increasing visual shape complexity, we see increasing entropy from Figs. 3(a)–3(h). Also, notice that entropy has some relationship to the symmetry of the curve, as well. The more symmetrical curves Figs. 3(a)–3(d) have lower shape information relative to the other curves. Again, this result is intuitive since symmetry implies repetition and thus duplication implies less information. Consider the two curves in Figs. 3(c) and 3(d). Both of these curves appear quite different perceptually, but both have equivalent symmetry. The entropy measure reflects this fact as it is also equivalent. The symmetry of both leads to equivalent shape information. The final comment concerns the magnitude of curvature. We note that in our formulation of entropy that the magnitude of curvature does not necessarily effect the amount of information that a shape contains. This notion initially seems counter intuitive since we would tend to believe that shapes with higher curvature would have more information. Higher curvature however is rather a level of degree and not a level of information. Locally, higher curvature points are similar to lower curvature ones when we view them from a microscopic scale-invariant perspective. A small circle in other words contains the same amount of information as a large circle where the difference is scale not information.

3. MANIFOLD 3D SURFACES

We now move our attention to manifold 3D surfaces. Curvature for surfaces is slightly more complex than curvature for planar curves. As with 2D, we again derive the following discussion from [11]. On a smooth surface S, we can define *normal curvature* as a starting point. Consider Fig. 4. The point p lies on an smooth surface S, and we specify the orientation of S at p with the unit-length normal N. We define S as a manifold embedded in $\mathbf{R}^3$. We can construct a plane Π_p that contains p and N such that the intersection of Π_p with S forms a contour α. As before, we can arbitrarily parameterize $\alpha(s)$ by arc length s where $\alpha(0) = p$ and $\alpha'(0) = T$. The normal curvature $\kappa_p(T)$ in the direction of T



(a) (b)

Fig. 4. Illustration of curvature for a surface.

is thus $\alpha''(0) = \kappa_p(T)N$. This single curvature $\kappa_p(T)$, however, does not specify the surface curvature of S at p.

We need to do a little more work since Π_p is not a unique plane. If we rotate Π_p around N, we form a new contour on S with its own normal curvature. For this infinite set, we can construct an orthonormal basis $\{T_1, T_2\}$. The natural choice for this basis is the tangent vectors associated with the extrema normal curvatures at p since they are always orthogonal [11]. These vectors $\{T_1, T_2\}$ are the *principal directions*. The curvatures $\kappa_p^1 = \kappa_p(T_1)$ and $\kappa_p^2 = \kappa_p(T_2)$ associated with these directions lead to the following relationship at p: $\kappa_p(T_\theta) = \kappa_p^1 \cos^2(\theta) + \kappa_p^2 \sin^2(\theta)$, where $T_\theta = \cos(\theta)T_1 + \sin(\theta)T_2$ and $-\pi \leq \theta < \pi$ is the angle to vector T_1 in the tangent plane. The extrema curvatures are known as the *principal curvatures* and completely specify the shape of S at p. Combinations of the principal curvatures lead to other definitions of surface curvature. Perhaps the most common is *Gaussian curvature*, which is the product of the principal curvatures $K_p = \kappa_p^1 \kappa_p^2$.

For our discrete formulation, we use Gaussian curvature because a simple formula—the angle excess formula—exists to compute Gaussian curvature from a mesh that approximates a smooth surface S. Consider Fig. 2(b). The angle ϕ_i is the wedge subtended by the edges of a triangle whose corner is at the vertex of interest. The angle excess is as follows:

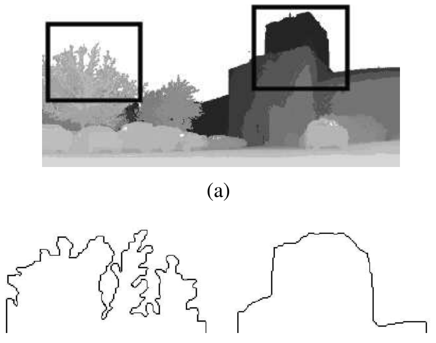
$$\Phi_j = 2\pi - \sum_i \phi_i \ , \tag{4}$$

where Φ_j is the angle excess for vertex j. The angle excess Φ_j has a direct relationship to Gaussian curvature K_p where we assume that vertex j approximates point p. Using a similar binning process as in the previous section, we can estimate the pdf for Φ and thus compute the shape information H for M bins.

4. EXPERIMENTAL RESULTS

We now investigate the behavior of the algorithm through both synthetic data and real data. To begin, the contours in Fig. 5 are extracted from the range image shown. As we would expect, the contour for the trees 5(b) has a higher shape information than the building 5(c). This result is similar to the results in [2] and is useful for sorting boundary contours of man-made and natural objects.

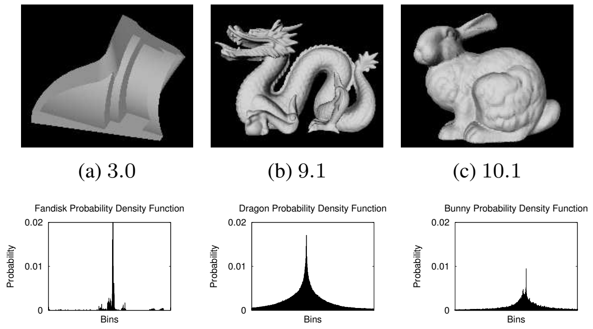
Figs. 6 through 8 show the experimental results for the 3D meshes. The entropy H_Δ below each mesh is in bits. The fandisk in Fig. 6(a) is a free-form CAD model with sharp edges and sophisticated surface curvature. This mesh has the least amount of shape information as most of the surface has constant curvature patches. The dragon in Fig. 6(b) and bunny in Fig. 6(c) have greater shape information values, due to the variation in features. Surprisingly, however, the bunny has the largest entropy value. Upon further inspection, we see why. The dragon is a symmetrical model with many features on one side duplicated on the other, and the features are repetitive as noted in the zoom view for the skin scales in Fig. 7(a). The bunny on the other hand is not quite as symmetrical and the skin in Fig. 7(b) has much greater variation across the whole model. These factors lead to more shape information. A more rigorous investigation is to examine the pdfs in Figs. 6(d)–6(f). If we consider the pdfs for the dragon and bunny in Figs. 6(e)–6(f). We see that the bunny has a larger variance. The more variation in the bunny model, as reflected in the pdf, results in the larger shape information value.



(a)

(b) 1.7 (c) 0.5

Fig. 5. Example 2D contours from a laser range image. Below each we show the shape information in bits. (a) Original range image. (b) Boundary contour extracted from (a) of tree line. (c) Contour for building.



(a) 3.0 (b) 9.1 (c) 10.1

(d) (e) (f)

Fig. 6. Example 3D meshes. Below each we show the shape information in bits. (a) Fandisk has 12K triangles (`http://research.microsoft.com/research/graphics/hoppe/`). (b) Dragon has 870K triangles. (c) Bunny has 2M triangles (both (b) and (c) `http://www.cc.gatech.edu/projects/large_models`). (d–f) Plots of the pdf estimates.

The final examples are in Fig. 8 where we have used a sheet-of-light range scanner and a collection of software tools to create meshes of actual objects. Intuitively, the entropy estimates match the shape complexity where the waterneck part in Fig. 8(a) has the least shape information and the crank in Fig. 8(c) has the most.

As with the curves in the prior section, a few caveats are important. The first caveat regards surfaces with boundaries. The models shown here are each water tight, which is to say they are

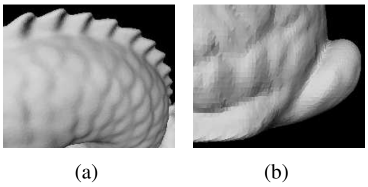
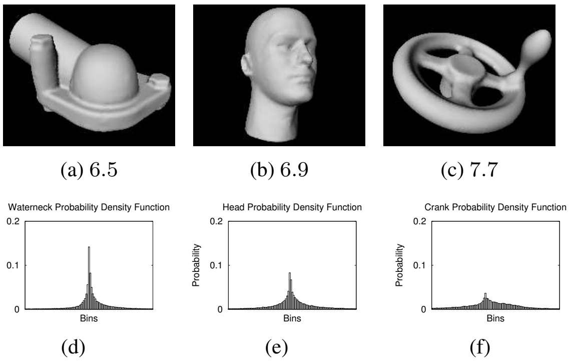

(a) (b)

Fig. 7. Zoom views of skin for dragon and bunny.


(a) 6.5 (b) 6.9 (c) 7.7

(d) (e) (f)

Fig. 8. Example 3D meshes from laser range data. Below each we show the shape information in bits. (a) Waterneck has 118K triangles. (b) Head has 80K triangles. (c) Crank has 94K triangles. (d–f) Plots of the pdf estimates, respectively.

surfaces without boundaries. For boundaries, we simply discard boundary vertices as we did with end points for curves. Another caveat regards the selection of the bin width. Since the bin width governs the probabilities of the pdf and ultimately affects the computation of entropy, we must use the same bin width for each model in order to validate comparison of entropy values. In other words, as we change the bin width, we can potentially change the entropy values and invalidate comparisons among models. For this paper, we use a common bin width $\Delta\Phi = 5 \times 10^{-5}$. We designate our entropy values H_{Δ} with a subscript Δ to emphasize this point.

5. CONCLUSIONS

In this paper, we have introduced a novel description of shape complexity for 2D image contours and 3D triangular meshes. Inspired by Shannon's concepts for information theory, we call our description *shape information*. We have proposed an algorithm for computing this metric based on curvature estimates for both discrete curves and surfaces. The unique contribution of our algorithm is that we compute a scalar metric that quantifies shape complexity. This metric, as demonstrated in Figs. 3, 6, and 8, agrees with the intuition of most human observers and how they might classify the relative complexity of these shapes. A concept derived from our proposed metric is that large curvature values do not necessarily increase shape information. Another trait of our metric is that shape redundancy, such as symmetry or repeating features, lowers the shape information of a curve or surface. Finally, a key assumption for our algorithm is uniform sampling. This constraint is a strong assumption and we thus direct our future investigations toward this topic.

6. ACKNOWLEDGEMENTS

This work is supported by the University Research Program in Robotics under grant DOE-DE-FG02-86NE37968, by the DOD/ TACOM/NAC/ARC Program, R01-1344-18, and by FAA/NSSA Program, R01-1344-48/49.

7. REFERENCES

[1] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, July,Oct. 1948.

[2] L. A. Oddo, "Global shape entropy: A mathematically tractable approach to building extraction in aerial imagery," in *Proceedings of the 20th SPIE AIPR Workshop*, 1992, vol. 1623, pp. 91–101.

[3] R. M. Cesar and L. Costa, "Application and assessment of multiscale bending energy for morphometric characterization of neural cells," *Review of Scientific Instruments*, vol. 68, no. 5, pp. 2177–2186, May 1997.

[4] T. Brinkhoff, H.-P. Kriegel, R. Schneider, and A. Braun, "Measuring the complexity of polygonal objects," in *Proceedings of the Third ACM International Workshop on Advances in Geographical Information Systems*, 1995, pp. 109–117.

[5] D. King and J. Rossignac, "Optimal bit allocation in compressed 3D models," *Computational Geometry: Theory and Applications*, vol. 14, pp. 91–118, 1999.

[6] G. Toussaint, "Efficient triangulation of simple polygons," *Visual Computer*, vol. 7, pp. 280–295, 1991.

[7] B. Chazelle and J. Incerpi, "Triangulation and shape complexity," *ACM Transactions on Graphics*, vol. 3, pp. 135–152, 1984.

[8] I. T. Young, J. E. Walker, and J. E. Bowie, "An analysis technique for biological shape I," *Info Control*, vol. 25, pp. 357–370, 1974.

[9] L. J. van Vliet and P. W. Verbeeck, "Curvature and bending energy in digitized 2D and 3D images," in *Proceedings of the Eighth Scandinavian Conference on Image Analysis*, 1993, vol. 2, pp. 1403–1410.

[10] N. R. Pal and S. K. Pal, "Entropic thresholding," *Signal Processing*, vol. 16, pp. 97–108, 1989.

[11] M. P. do Carmo, *Differential Geometry of Curves and Surfaces*, Prentice–Hall, Inc., Englewood Cliffs, NJ, 1976.