Embedding Gestalt Laws in Markov Random Fields

Song-Chun Zhu

Abstract—The goal of this paper is to study a mathematical framework of 2D object shape modeling and learning for middle level vision problems, such as image segmentation and perceptual organization. For this purpose, we pursue generic shape models which characterize the most common features of 2D object shapes. In this paper, shape models are learned from observed natural shapes based on a minimax entropy learning theory [31], [32]. The learned shape models are Gibbs distributions defined on Markov random fields (MRFs). The neighborhood structures of these MRFs correspond to Gestalt laws—colinearity, cocircularity, proximity, parallelism, and symmetry. Thus, both contour-based and region-based features are accounted for. Stochastic Markov chain Monte Carlo (MCMC) algorithms are proposed for learning and model verification. Furthermore, this paper provides a quantitative measure for the so-called *nonaccidental statistics* and, thus, justifies some empirical observations of Gestalt psychology by information theory. Our experiments also demonstrate that global shape properties can arise from interactions of local features.

Index Terms—Gestalt laws, perceptual grouping, shape modeling, Markov random field, maximum entropy, shape synthesis, active contour.

1 Introduction and Motivations

In psychology, it has long been evident that early human vision strongly favors certain shapes and configurations over others without high level identification. Many theories have been proposed to account for this phenomenon, among which Gestalt psychology is the most influential one. In Gestalt psychology, an enormous number of generic laws have been identified for grouping *parts* to *whole* [17], for example,

"proximity, continuity, colinearity, cocircularity, parallelism, symmetry, closure, familiarity."

These laws are supposed to be coordinated by the law of Pragnanz [17]:

"of several geometrically possible organizations that one will actually occur which possesses the best, simplest and most stable shape (p.138)."

But, what is meant by a good, simple and stable shape? Gestalt psychologists explained it in terms of "field forces" by analogy to field theories of gravity and electricity, but it is unclear what the "field forces" really are. Even worse, Gestalt psychology only provides a descriptive theory, it does not specify a computational process for achieving the percept from *parts* to *whole*. Although Gestalt laws have been successfully utilized in many perceptual organization algorithms [22], [24], a rigorous mathematical theory has yet to be found.

Besides Gestalt psychology, there are two other theories for perceptual organization. One is the *likelihood principle* [9], which assigns a high probability for grouping two elements, such as line segments, if the placement of the two elements has a low likelihood of resulting from *accidental arrangement* [21], [22]. The third theory is the simplicity or *minimum description length principle* [10], which states that perceptual organization should achieve the shortest coding length by exploring shape symmetry, etc. We argue that the key to understanding the psychophysical phenomena is to pursue the "true" underlying probability distribution for the ensemble of object shapes (or configurations) in a given application domain. With this probability distribution, the three theories can be well unified for the following reasons.

1. According to Shannon's coding theory, a "true" probability yields the minimum expected coding length for the ensemble of shapes (or configurations).[1] Since the shape distribution accounts for the complexity and frequency of configurations occurring in nature, it gives the genuine likelihood probability for groupings. As we will see in a later section, *nonaccidental* statistics can also be measured using this distribution.
2. The importance of Gestalt laws can be quantified by studying statistical regularities in the shape ensemble. These Gestalt laws should be reflected in the structures of the probability distribution, provided they are indeed effective.

In two previous papers, the author, in collaboration with Wu and Mumford, has studied probability models for textures and natural images by a minimax entropy principle [32], [31]. This paper applies the minimax entropy principle to learning probability distributions of 2D shapes. In particular, we are interested in simple closed curves which

• *The author is with the Department of Computer and Information Science, The Ohio State University, Columbus, OH 43210. E-mail: szhu@cis.ohio-state.edu.*

Manuscript received 25 Mar. 1998; revised 2 June 1999. Recommended for acceptance by K.V. Mardia. For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 107645.

1. The coding length should also include the code book, thus simpler models are favored in model selection.

are non-self-intersect region boundaries in 2D images projected from 3D objects. A basic assumption is that there exists a true underlying distribution for 2D object contours in a given application, and shape modeling is posed as a statistical inference problem.

We shall address the following questions in this paper:

1. Studies in neurosciences have shown that statistics of visual environment (i.e., the ecological effects) play key roles on the functions of both individual cells and neural systems [1].[2] Is there any ecological evidence—embedded in the statistics of realistic shapes—for Gestalt laws and early human perception of 2D shapes?
2. How can various Gestalt laws be integrated into a single probability measure?
3. What are the structures and forms of generic probability distributions for 2D shapes?
4. How do we sample shape distributions? What are the typical shapes sampled from such distributions? To what extent can real shapes—with global properties—be characterized by merely a few locally defined Gestalt laws?

We start with exploring statistics of features extracted from observed real shapes and, then, a shape distribution is learned such that it reproduces the observed statistics while having the maximum entropy. The learned shape models are Gibbs distributions on Markov random fields. The variables of these MRFs are the coordinates of points along the contours and the structures of the MRFs correspond to Gestalt laws such as proximity, colinearity, cocircularity, parallelism, and symmetry. The learned shape models are verified by Markov chain Monte Carlo sampling. The paper also provides a quantitative measure for *nonaccidental statistics* by comparing observed statistics and statistics of randomly sampled shapes. We demonstrate that global shape properties could arise through propagation of local interactions in Markov random fields.

The paper is organized as follows: We start with reviewing existing theories for shape modeling in Section 2 and discuss basic issues in shape modeling in Section 3. Section 4 discusses feature extraction and Section 5 demonstrates experiments on statistics of animate shapes. Section 6 presents a maximum entropy theory for probability learning. Section 7 discusses issues of designing stochastic sampling processes in a shape space and describes experiments of sampling a uniform distribution of shape. Section 8 discusses feature selection by maximizing nonaccidental statistics. Section 9 is devoted to experiments on shape learning and sampling. Finally, we conclude the paper by a discussion in Section 10.

2. For example, there are considerable functional differences of neurons in retina between rabbit and fish and these differences are supposed to be explained by statistical properties of their living environments [1]. Here, we are interested in knowing why early human vision is sensitive to Gestalt features.

2 Previous Theories in 2D Shape Modeling and Perceptual Grouping

Two-dimensional shapes have been studied intensively in many disciplines, ranging from pure mathematics, statistics, psychology, to computer vision. These studies are divided into four areas:

1. a statistical theory of shape,
2. deformable templates and models,
3. nonaccidental properties and perceptual grouping, and
4. active contour models.

In this section, we shall review these theories.

2.1 Statistical Theory of Shape

The term "theory of shape" was coined by Kendall in 1977 (see [15]). In Kendall's theory, a shape is defined as a set of k points in m dimensions. Thus, a shape is naturally represented as an $m \times k$ matrix. To "filter" out effects from translation, scaling, and rotation, Kendall first defines one point as the origin, which reduces the matrix to a size of $m \times (k-1)$, and then he normalizes the shape by setting the sum of the squared elements of the matrix to one. Thus, all his shapes live on the surface of a unit sphere in $m \times (k-1)$ dimensions, which he called the *preshape sphere*. The real shape space, denoted by $\sum_m^k$, is the quotient of the preshape sphere by some transformation groups, e.g., $SO(m)$. Then, each $SO(m)$ equivalence class is viewed as a single point—a real shape [15]. For example, assuming independent and identical uniform distribution for the k points, Kendall makes inference about the statistical significance of colinearity in a given point set and of the size of a hole in a Delaunay tessellation of Galaxy [15] to answer queries about how likely the colinearities and holes are not accidental arrangements by i.i.d. uniform distributions.

The theory of shape has been also studied by Bookstein in morphometrics—a discipline studying deformations and variabilities of biological organisms [2]. Here, the point set is the collection of unique landmarks which correspond biologically from object to object. Recently, Mardia and Dryden have studied the theory of shape in the context of image analysis with interesting results [23].

2.2 Deformable Templates—High Level Vision

Another elegant theory for shape modeling was pioneered by Grenander in a discipline which he named *pattern theory* [7]. In pattern theory, shape primitives are selected, such as edge segments, and these primitives are arranged in prespecified configurations, such as a circular graph. Then, transformation groups (rotation, scaling, etc.) act on these configurations to account for both global and local deformations.

For example, the contour of a human hand is represented as a ring of linelets [8], and a global transform specifies the location, size, and orientation of the hand, and local transforms define relative orientations of fingers, and so on. Then, a probability distribution is defined on these groups. This shape modeling scheme has been used in representing leaves and brain mapping [8], [7].

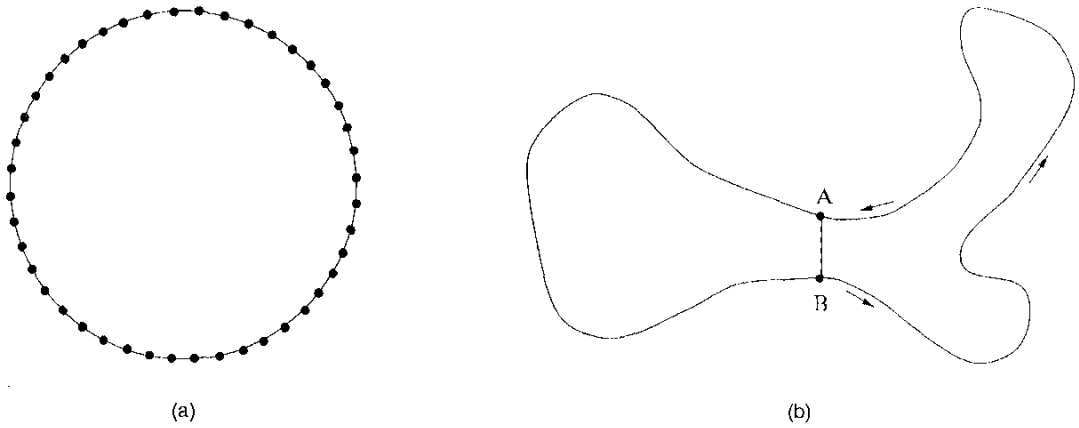


Fig. 1. (a) A 1D Markov random field where the nodes represent random variables for positions of contour points. (b) Node A is spatially adjacent to point B, but it is far away from B in the circular neighborhood of (a).

In computer vision, deformable templates were proposed by Yuille in studying shapes of human eyes, mouth, and eyebrows, etc. [28]. In Yuille's templates, shapes consist of piecewise conics and deformations are modeled based on coefficients. Other deformable models of shapes are defined in terms of basis functions, such as B-splines, sine waves [4], and implicit polynomials [16], and super quadrics [26]. A more general template model for flexible object is studied in the FORMS system [30], where object shapes are defined in graphs computed from medial axes and deformations are specified by PCA modes learned from animal shapes. Other shape models are represented in Bayes networks [5]. In fact, the statistical theory of shape studied in (Bookstein 1986, Mardia and Dryden 1989) can also be considered as shape templates with points being primitives.

2.3 Nonaccidental Properties and Perceptual Organization—Middle Level Vision

The problem of perceptual organization was formally studied by Lowe in 1985 [21]. Lowe pointed out that the goal of grouping is to identify features that are likely to have arisen from some scene properties rather than accidental arrangements. Lowe proposed some measures that account for how nonaccidental the arrangement is for each individual grouping feature, such as, colinearity and parallelism. This theory has been used in grouping and recognizing rigid objects with interesting results [22], [24]. However, it doesn't provide a rigorous probability measure for shapes and it is unclear how to **decorrelate** multiple shape features in computing the significance of a nonaccidental arrangement. Similar to Lowe's theory, nonaccidental properties are also studied for convexity of a sequence of line segments by Jacobs in 1992 [11].

In the literature of perceptual grouping, there are no rigorous attempts for learning and verifying probabilities models from real shapes.

2.4 Active Contour Models—Low Level Vision

In early vision tasks, generic shape models are found to be useful. One important generic shape model is the active contour model (SNAKE) [14], and the internal energy in a SNAKE model implicitly defines a prior probability distribution of a curve $\Gamma(s)$:

$$p(\Gamma) = \frac{1}{Z}\exp\left\{-\int \alpha|\dot{\Gamma}(s)|^2 + \beta|\ddot{\Gamma}(s)|^2 ds\right\}, \qquad (1)$$

where Z is a normalization constant, $\dot{\Gamma}(s)$ and $\ddot{\Gamma}(s)$ are the first and the second derivatives of the curve, and s is usually the arc-length.

An explicit probability model for open curves, called the Elastica model, was first derived from Brownian motion by Mumford in 1994 [25] and it was also studied by Williams and Jacobs in 1997 [27]:

$$p(\Gamma) = \frac{1}{Z}\exp\left\{-\int \beta + \alpha\kappa^2(s) ds\right\}, \qquad (2)$$

where $\kappa(s)$ is the curvature of the curve.

In discrete cases, both (1) and (2) are defined on interactions between nodes in a local neighborhood, illustrated in Fig. 1a. Despite the wide applications of active contour models in computer vision, they have two main problems.

First, they do not characterize region-based information. For example, in the shape shown in Fig. 1b, node A is close to node B in distance, whereas they are far away from each other along the boundary.

Second, potential functions are often manually designed, whereas it is desirable to learn them from data.

In summary, Sections 2.4, 2.3, and 2.2 briefly review existing theories in the low, middle, and high level vision respectively. It is still unclear how these theories can be combined for image understanding in a consistent manner and the incompatibility of these models (or theories) in the three levels poses a major obstacle for building sophisticated vision systems.

In this paper, we shall study a mathematical framework of 2D shape modeling for middle level vision problems, such as image segmentation and perceptual organization. Motivated by the above discussion, we pose two criteria in our models. 1) These shape models should be generic, and thus characterize the most common features of 2D shapes.

2) They should be compatible with low level representations, such as raw images and edge maps, and the high level descriptions, such as deformable templates.

3 Shape Spaces and Probability Measures

Let $\Gamma(s)$ be a simple and closed contour on 2D plane,[3] where $s \in [0,1]$ is the arc-length and $\Gamma(0) = \Gamma(1)$. Fixing the resolution, $\Gamma(s)$ is discretized into a polygon of N vertices:

$$\Gamma = ((x_0, y_0), (x_1, y_1), \ldots, (x_{N-1}, y_{N-1})), \quad \Gamma \in \Omega_\Gamma \subset R^{2N},$$

where Ω_Γ is the space of shapes satisfying the following hard constraints.

Constraint I: $\Gamma \in \Omega_\Gamma$ is closed and non-self-intersecting.
Constraint II:

$$\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \in [ds - \epsilon, ds + \epsilon]$$
$$i = 0, 1, \ldots, N-1.$$

In Constraint II, the distance of sequential nodes is allowed to vary slightly because of finite precision of lattice in computer implementation. We shall discuss the effect of ϵ when we discuss a Gibbs sampler for sampling shapes in Section 7. The structures of Ω_Γ perhaps are too complicated to analyze explicitly.

On Ω_Γ, a Lebesgue measure is well-defined:

$$P(\Gamma) = p(\Gamma) dx_0 dy_0 dx_1 dy_1 \cdots dx_{N-1} dy_{N-1}, \tag{3}$$

with $p(\Gamma)$ being the density that we shall define in later sections.

Following Kendall's treatment [15], we shall "filter out" some effects of transforms and define a "genuine" space of shape Ω_Γ^*. Ω_Γ^* is the quotient space of Ω_Γ under 2D translation and planar rotation, as well as cyclic permutation of nodes. Each "genuine" shape in Ω_Γ^* is the projection of an *equivalence class* in Ω_Γ under similarity transforms and cyclic permutation. The reasons for such treatment are twofold.

First, object silhouettes are taken into images at arbitrary distances, locations, and orientations. Therefore, a shape model should be invariant to translation, rotation, and scale.

Second, for generic shape models in middle level vision, we assume that any features should have the same chance to appear in any location s in the contour. Thus, the shape model is invariant to cyclic permutation of nodes.

In the rest of the paper, $p(\Gamma)$ is assumed to be **homogeneous** with respect to s and $p(\Gamma)$ is defined on relative positions and orientations. Furthermore, Γ is normalized to have unit length. Thus, the density $p(\Gamma)$ is invariant to translation, rotation, scale, and cyclic permutation. As we shall discuss in Section 10, nonhomogeneous properties can be built at high level representation.

Now, we need to define a probability measure $\nu(\Gamma)$ on Ω_Γ^*, which takes the sum of the Lebesgue measure over an equivalence class in Ω_Γ. If we only consider translation and cyclic permutation transforms, the equivalence classes all have the same size, therefore, $\nu(\Gamma)$ is simply the Lebesgue measure multiplied by a constant.[4] When rotation is added, the sizes of the equivalence classes may vary because of rotational symmetry in shapes. However, this should not be a concern in computer vision because of the following reason.

Suppose Γ is a rotationally symmetric polygon, for example, Fig. 2 displays an N-gon with degree $d = 4$ rotational symmetry. Suppose we discretized $[0, 2\pi]$ into $1,000$ angles, the equivalence class for this shape includes only 997 distinct shapes, while a nonsymmetric shape has $1,000$ distinct shapes in its equivalence class. With precise discretization, the size variation of equivalence classes is negligible except for shapes, e.g., a perfect circle. However, because we are studying natural animate shapes in this paper, the probability measure for shapes which have high degrees of rotational symmetry is **exponentially** small, that is, these highly symmetric shapes have measure 0. For example, none of the observed and synthesized shapes in the later section demonstrate any rotational symmetry at all!

In a given application domain, the ensemble of shapes is assumed to be governed by a true underlying distribution $f(\Gamma)$ on Ω_Γ (or Ω_Γ^* with no practical difference). A set of 2D object contours, $\{\Gamma_i^{\text{obs}}, i = 1, 2, \ldots, M\}$, are observed as independent samples from $f(\Gamma)$ (see examples in Fig. 4). The objective of shape modeling is to learn a probability model $p(\Gamma)$ as an estimation of $f(\Gamma)$. In computer vision, $p(\Gamma)$ is often called the *prior model*, and can be used for image segmentation, contour tracing, shape completion, and so on in the Bayesian framework.

4 Feature Selection

To learn a shape model $p(\Gamma)$, we start with exploring both contour-based and region-based features on Γ. We are interested in some simple Gestalt laws: colinearity, cocircularity, proximity, parallelism, and symmetry, each of which is measured by a continuous function.

Fig. 3a shows a dog shape Γ, which is decomposed into a number of sequential linelets $\ell(s)$. $\ell(s)$ includes attributes $(x(s), y(s), \theta(s))$ for the center coordinates and orientation. Each linelet is represented by a node on a circle in Fig. 3b. The circular connection defines a *random field* as in Fig. 1a, where $x(s), y(s), \theta(s)$ are random variables at each site s and the neighborhood structures of the random field are shown in Fig. 3b (to be discussed later in this section).

Now, we discuss how shape features are extracted from random fields.

At each location s,[5] we measure a set of functions $\phi^{(\alpha)}(s)$ with $\alpha = 1, 2, 3, \ldots$ being the index of shape features. By analogy to feature extraction using Gabor filters in 2D images, $\phi^{(\alpha)}(s)$ is the "response" for a "shape filter" at location s.

We first define two contour-based functions, the curvature $\kappa(s)$ and derivative of curvature:

3. In the rest of the paper, the word shape refers to a simple, closed, and non-self-intersecting contour.

4. With a slight modification, we can define shapes in a finite, but large enough, 2D rectangular domain instead of an infinite plane; then the equivalence classes have finite size.

5. For simplicity of notation, we derive the shape models in continuous representation and, then, we convert it to the discrete lattice at other places.

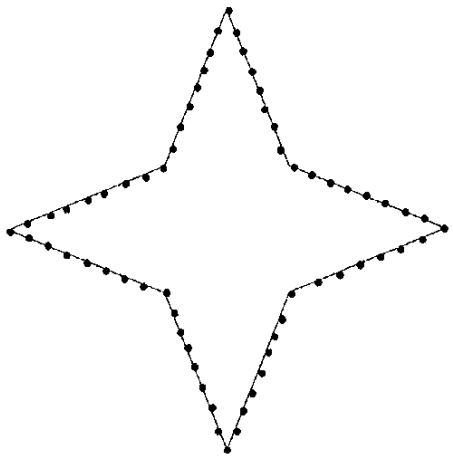


Fig. 2. An N-side polygon with rotational symmetry of degree $d = 4$.

$$\phi^{(1)}(s) = \kappa(s) = \frac{d\theta(s)}{ds}$$

and

$$\phi^{(2)}(s) = \nabla\kappa(s) = \frac{d^2\theta(s)}{ds^2}, \quad \forall s \in [0,1].$$

Obviously, $\phi^{(1)}(s) = 0$ means that two adjacent linelets are *colinear* and $\phi^{(2)}(s) = 0$ means that three sequential linelets are *cocircular*. Other contour-based shape filters can be defined in the same way.

In the following, we proceed to define region-based properties.

It is well-known in computer vision that real object shapes—resulting from processes of accretion—have natural descriptions in terms of medial axes [20], [30]. For example, a dog has a skeleton and many elongated parts. As shown in Fig. 3a, linelets at the two sides of a limb or a torso are parallel or symmetric with respect to a medial axis. It is also evident in psychophysical experiments that early human vision is sensitive to medial axis [18], [3]. This observation has its deep roots in physiology, where experiments have demonstrated that some neurons in the primary visual cortex (V1) of monkeys compute symmetric axes as soon as they detect edge elements [19].

To capture region-based features of a curve $\Gamma(s)$, the author has defined a symmetry mapping function $\psi(s)$ [33],

$$\psi : [0,1] \to [0,1], \qquad \psi(s) = t \Longleftrightarrow \psi(t) = s.$$

Fig. 3 displays the mapping function ψ for a dog shape. As shown in Fig. 3b, ψ divides the circular domain $[0,1]$ into 18 disjoint intervals:

$$\mathcal{I}_i = (s_{i0}, s_{i1}) \subset [0,1], \quad \mathcal{I}_i \cap \mathcal{I}_j = \emptyset, i \neq j.$$

ψ establishes a piecewise continuous mapping $(s, \psi(s))$ between these intervals under a hard constraint that these mapping line segments don't cross each other.

Intuitively, each pair of intervals represent an elongated part of a shape. For example, the mapping function ψ for the dog shape, shown in Fig. 3, has nine pairs of intervals for the nine parts of the dog. Note that, in Fig. 3a, some linelets are matched more than once due to the effect of discretization. As illustrated in Fig. 3b, the mapping function ψ defines a new neighborhood for nodes on shape Γ and opens "communication channels" for linelets across regions.

Now, we briefly explain how ψ is computed.

The mapping function $\psi(s)$, as well as intervals, is computed by minimizing an energy functional, which is defined to enforce the following two aspects:

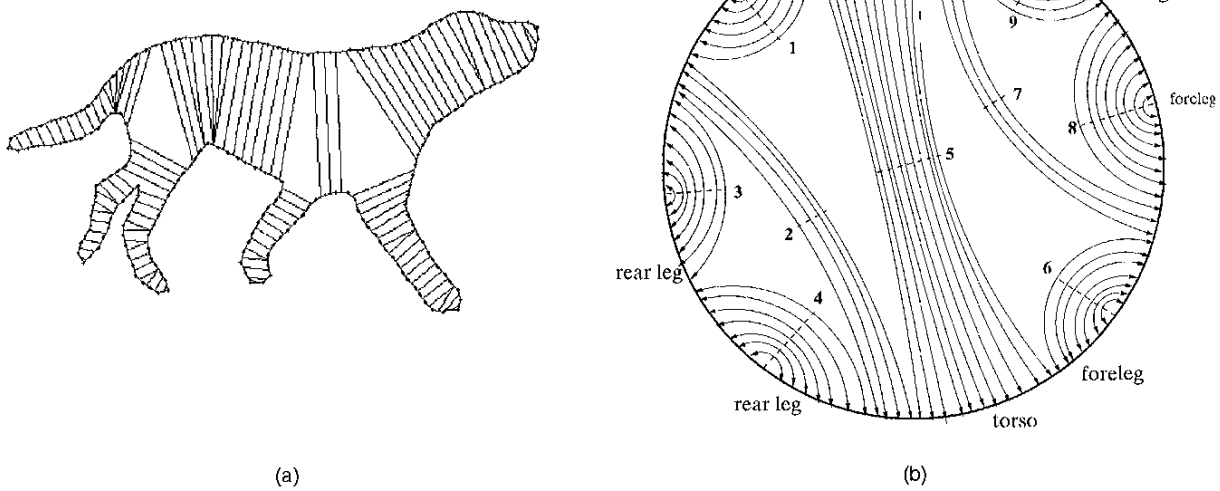


Fig. 3. (a) A dog shape with region-based correspondence detected. (b) An abstract planar *adjacency graph* representing the neighborhood structures of the linelets in the dog shape.

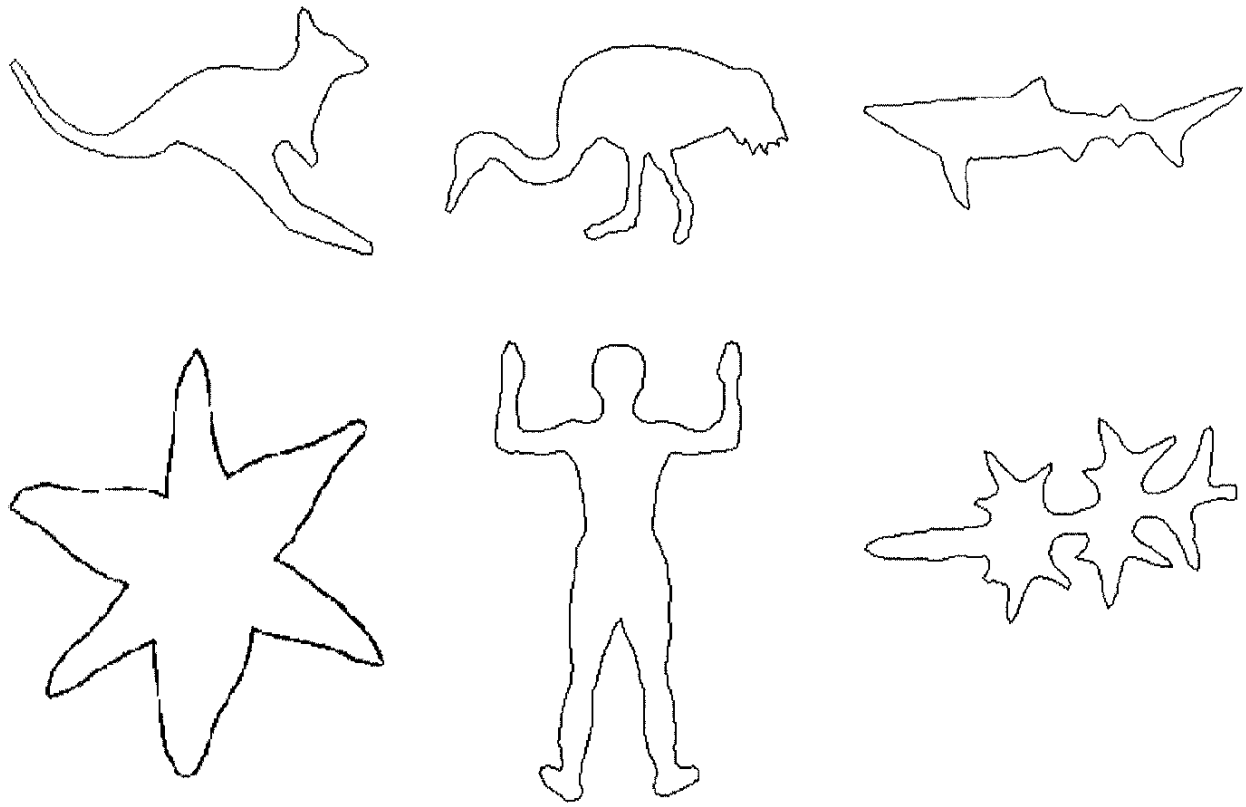


Fig. 4. Examples of the observed natural shapes.

1. Two matched linelets $\ell(s)$ and $\ell(\psi(s))$ should be as close, parallel, and symmetric to each other as possible.
2. The number of intervals (or discontinuities of $\psi(s)$) in $[0, 1]$ should be as small as possible.

The optimal ψ is computed by a stochastic algorithm (Gibbs sampler) based on local neighborhood in a Markov random field and the medial axis is then computed based on ψ. Details of the definition of the energy functional and the algorithm is referred to a companion paper [33].

Let $r(s)$ be the distance between two matched linelets $\ell(s)$ and $\ell(\psi(s))$. $r(s)$ is divided by the length of the curve, so it is well-normalized with respect to scale. We call $r(s)$ the "rib" length. Then, we define three region-based shape features,

$$\phi^{(3)}(s) = r(s),$$
$$\phi^{(4)}(s) = \nabla r(s) = \frac{dr(s)}{ds},$$
$$\phi^{(5)}(s) = \nabla^2 r(s) = \frac{d^2 r(s)}{ds^2}.$$

$\phi^{(4)}(s) = 0$ means two linelets are parallel to each other, $\phi^{(5)}(s) = 0$ means two pairs of linelets are symmetric to each other. $\phi^{(3)}(s)$ measures the proximity between two linelets across a region, as we discussed in Fig. 1b.

It is easy to see that all functions $\psi^{(\alpha)}(s), \alpha = 1, 2, 3, 4, 5$ are continuous measures for the shape properties and the Gestalt laws and they are invariant to translation, rotation, and scaling transforms in a 2D plane.

Another interesting shape feature is

$$\phi^{(6)}(s) = \begin{cases} 1 & \text{if } \psi(s) \text{ is discontinuous at s} \\ 0 & \text{otherwise.} \end{cases}$$

This measures the number of breaks and branches (or parts) of an object.

5 Statistics of Animate Shapes

In this section, we shall discuss how to compute statistics of natural 2D shapes for a set of features $\Phi = \{\phi^{(\alpha)}, \alpha = 1, 2, \ldots\}$.

We collect a set of $M = 22$ shapes of animate objects, $\{\Gamma_i^{\text{obs}}, i = 1, 2, \ldots, 22\}$, six of which are displayed in Fig. 4. These shapes are assumed to be independent samples from a true distribution $f(\Gamma)$. We are interested in animate shapes because they have richer flexibilities and variations than human-made objects.

These animate shapes are acquired at various resolutions from 2D images with perimeter being L_i pixels long for $i = 1, 2, \ldots, 22$. A shape observed at a high resolution contains rich information. Γ_i^{obs} is represented by $N_i = \frac{L_i}{c}$ nodes or linelets, i.e., a polygon of N_i sides. Generally speaking, the side length c of a polygon should be as small as possible to obtain a good approximation to the continuous curve; on the other hand, c it should not be too small,

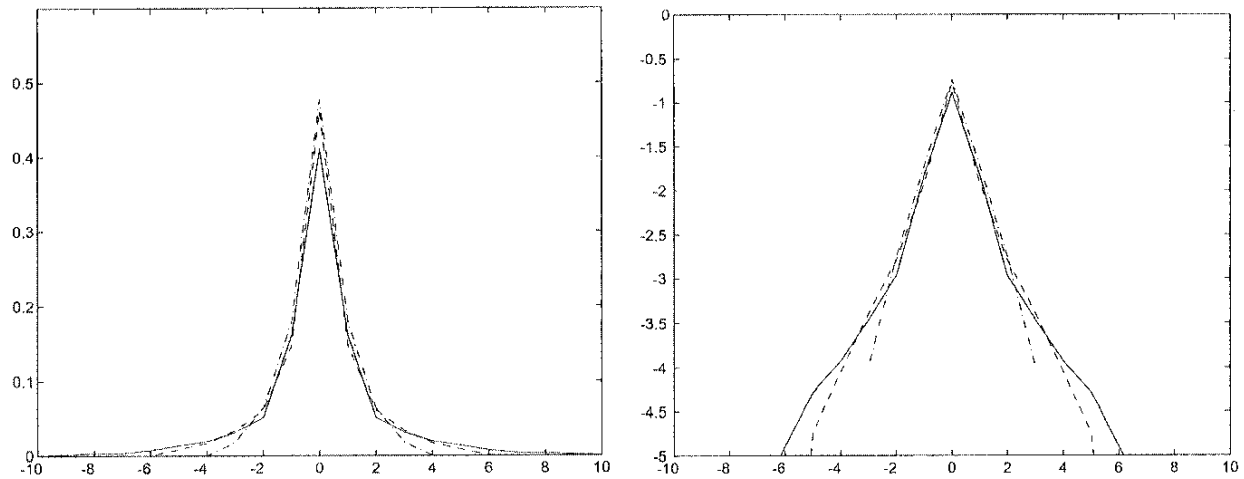


Fig. 5. (a) The histograms of $\kappa(s)$ averaged over 22 animate objects at scale 0 (solid curve), scale 1 (dashed curve), and scale 2 (dash-dotted curve), the horizontal axis is $\kappa(s)$ with unit $dz = \frac{\pi}{16\times 200}$. (b) The logarithm of curves in (a).

otherwise features, such as $\kappa(s)$, cannot be computed reliably. A good compromise is to make c as big as possible while there are no noticeable artifacts of polygon approximation to human perception. We choose $c = 6$ pixels in this paper. The arc length for each linelet in Γ_i^{obs} is $ds = \frac{1}{N_i}$, therefore, all shapes are normalized to the same scale.

In this paper, the statistics are extracted as empirical histograms of features. By homogeneity assumption, the histogram for $\phi^{(\alpha)}(s)$ on Γ_i^{obs} is

$$H^{(\alpha)}(\Gamma_i^{\text{obs}}, z) = \int \delta(z - \phi^{(\alpha)}(s))ds, \quad \alpha = 1, 2, \ldots, 6, \quad \forall s.$$

In the above definition, z is a continuous variable for the feature, e.g., $H^{(0)}(\Gamma_i^{\text{obs}}, 0)$ is the number of points on Γ_i^{obs} that have zero curvature. $\delta()$ is the Dirac delta function with unit mass at zero and $\delta(x) = 0$ for $x \neq 0$. For a discretized curve, we have

$$H^{(\alpha)}(\Gamma_i^{\text{obs}}, z) = \frac{1}{N_i}\sum_{j=1}^{N_i} \delta(z - \phi^{(\alpha)}(s_j)), \quad \alpha = 1, 2, \ldots, 6.$$

We further compute average histograms for the M discretized curves

$$\mu_{\text{obs}}^{(\alpha)}(z) = \frac{1}{N_1 + N_2 + \ldots + N_M}\sum_{i=1}^{M} N_i H^{(\alpha)}(\Gamma_i^{\text{obs}}, z), \quad \alpha = 1, 2, \ldots, 6. \tag{4}$$

If $\sum_{i=1}^{M} N_i$ is big enough, then $\mu_{\text{obs}}^{(\alpha)}(z)$ is a close estimation of the marginal distribution of the true model $f(\Gamma)$, i.e.,

$$\mu_{\text{obs}}^{(\alpha)}(z) \approx \mu^{(\alpha)}(z) = \int f(\Gamma)\delta(z - \phi^{(\alpha)}(s))d\Gamma, \quad \forall z, \forall s, \quad \forall \alpha.$$

In the rest of the paper, we assume that $\mu_{\text{obs}}^{(\alpha)}(z) = \mu^{(\alpha)}(z), \alpha = 1, 2, \ldots, 6$.

To study how the observed histograms change with the scales (resolutions) of natural shapes, we subsample each observed shape once and twice, and generate two new sets of observed shapes at scale 1 and scale 2. A shape at scale $i+1$ has only one-half of the nodes of the shape at scale i.

Fig. 5a plots the observed histograms $\mu_{\text{obs}}^{(1)}$ for $\phi^{(1)}$ averaged over 22 animate shapes at scale 0 (solid curve), scale 1 (dashed curve), and scale 2 (dash-dotted curve), respectively, and Fig. 5b plots the logarithms of these curves in Fig. 5a.

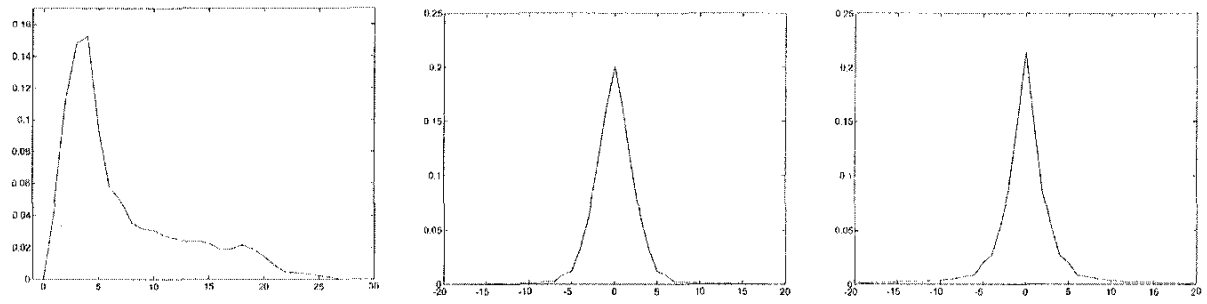
In our experiments, we flipped the 22 observed shapes horizontally to double the observed dataset to 44 shapes for more robust estimation of histograms because we should have the same chance to observe an animate object from both sides. As a result, the histograms for $\phi^{(\alpha)}, \alpha = 1, 2$ are perfectly symmetric with a peak at zero. This result may not be true if we had not had flipped the shapes. This treatment means that the probability model should also be invariant to a flip transform.

Fig. 5 demonstrates two interesting properties. First, the middle part of $\mu_{\text{obs}}^{(1)}$ is close to an exponential curve, but $\mu_{\text{obs}}^{(1)}$ has heavier tails. Second, unlike the scale invariant properties found in filter responses of natural images [31], the histogram of curvature is only approximately invariant to scales in these observed shapes. This is mainly because the subsampling procedure smoothes the curve faster at high curvature segments than at low curvature segments. As a result, the curvature histograms at scale $i+1$ have lighter tails and sharper peak near zero than histograms at scale i.

The average histogram $\mu_{\text{obs}}^{(2)}$ of $\phi^{(2)}$ is very close to the curvature histogram. Currently, it is unclear why it is so. We choose $\mu_{\text{obs}}^{(1)} = \mu_{\text{obs}}^{(2)}$ to be the observed curvature histogram averaged over scale 0 and scale 1 for robustness.

We also compute histograms $\mu_{\text{obs}}^{(\alpha)}, \alpha = 3, 4, 5$ for the region-based features, $\phi^{(3)}(s) = r(s)$, $\phi^{(4)}(s) = \nabla r(s)$, $\phi^{(5)}(s) = \nabla^2 r(s)$. These computed histograms are shown in Fig. 6.

Histograms for $\kappa(s), \nabla\kappa(s), \nabla r(s), \nabla^2 r(s)$ are not Gaussian distributions and they all have a sharp peak at zero,



(a) (b) (c)

Fig. 6. The observed histograms averaged over 22 animate shapes for region-based features. (a) Histogram of $r(s)$. (b) Histogram of $\nabla r(s)$. (c) Histogram of $\nabla^2 r(s)$.

indicating the statistical significance of colinearity, cocircularity, parallelism, and symmetry in natural shapes. The histogram of $r(s)$ has peak near zero, which implies important correlations and proximity of points across regions.

6 Shape Modeling by Maximum Entropy

This section derives shape models which account for the observed statistics.

6.1 From Empirical Histograms to Gibbs Distributions

Given features $\Phi = \{\phi^{(\alpha)}, \alpha = 1, 2, \ldots\}$ and the marginal distributions of $f(\Gamma)$ $\{\mu^{(\alpha)}_{\text{obs}}(z), \alpha = 1, 2, \ldots\}$, a model $p(\Gamma)$ should reproduce these marginal distributions. Thus, as far as the features in Φ are concerned, $p(\Gamma)$ cannot be distinguished from the true model $f(\Gamma)$. Intuitively, if we extract enough number of important features, $p(\Gamma)$ will be a close estimation of $f(\Gamma)$.

Let Ω_p be the set of all probability distributions $p(\Gamma)$ which can reproduce the observed statistics

$$\Omega_p = \left\{ p(\Gamma) \mid \int p(\Gamma)\delta(z - \phi^{(\alpha)}(s))d\Gamma = \mu^{(\alpha)}_{\text{obs}}(z) \quad \forall s, \forall z, \forall \alpha \right\}.$$

From Ω_p, a maximum entropy (ME) [12] distribution is chosen for it has the least bias in the unconstrained dimensions. Thus, a shape model is

$$p^*(\Gamma) = \arg\max - \int p(\Gamma) \log p(\Gamma) d\Gamma,$$

subject to constraints

$$\int p(\Gamma)d\Gamma = 1 \tag{5}$$

$$\int p(\Gamma)\delta(z - \phi^{(\alpha)}(s))d\Gamma = \mu^{(\alpha)}_{\text{obs}}(z) \quad \forall s, \forall z, \forall \alpha. \tag{6}$$

Solving the constrained optimization problem by Lagrange multipliers and calculus of variations, we obtain the following ME distribution:

$$p(\Gamma; \Phi, \Lambda) = \frac{1}{Z} \exp\left\{ -\sum_{\alpha=1}^{k} \int \lambda(\phi^{(\alpha)}(s))ds \right\}, \tag{7}$$

or, equivalently,

$$p(\Gamma; \Phi, \Lambda) = \frac{1}{Z} \exp\left\{ -\sum_{\alpha=1}^{k} \int \lambda(z) H^{(\alpha)}(\Gamma, z)dz \right\}. \tag{8}$$

In the above equations, Z is the normalization constant and it is also called the partition function $Z = Z(\Phi, \Lambda)$ in physics. $\Lambda = (\lambda^{(1)}(), \lambda^{(2)}(), \ldots, \lambda^{(k)}())$ are Lagrange multipliers, or potential functions. Since the constraints are imposed for continuous variables z and s, $\lambda^{(\alpha)}()$ is a continuous function. Because of the homogeneity assumption, $\lambda^{(\alpha)}(), \alpha = 1, 2, \ldots, k$ are independent of s and, therefore, they are one-dimensional.

It is easy to see that the active contour models in (1) and (2) are special examples of (7) with Φ and Λ specified.

6.2 Estimation and Computation

In model $p(\Gamma; \Phi, \Lambda)$, the Lagrange multipliers (or the potential functions) have yet to be solved from the constraint equations (5) and (6). In practice, due to the complexity of Φ and the shape space Ω_Γ, we can only compute Λ numerically. Our method for computing Λ has been successfully applied to texture modeling and prior learning in [32], [31], and we describe it briefly in this section.

In practice, a histogram $H^{(\alpha)}(\Gamma, z)$ is discretized into m bins, represented by a vector $H^{(\alpha)}(\Gamma) = (H^{(\alpha)}_1, H^{(\alpha)}_2, \ldots, H^{(\alpha)}_m)$. $\lambda^{(\alpha)}()$ is also approximated by a piecewise constant function, represented by a vector $\lambda^{(\alpha)} = (\lambda^{(\alpha)}_1, \lambda^{(\alpha)}_2, \ldots, \lambda^{(\alpha)}_m)$. If m is large enough, $\lambda^{(\alpha)}$ is a close approximation to $\lambda^{(\alpha)}()$. In general, m may vary with features.

We replace the integration in (8) by an inner product, therefore,

$$p(\Gamma; \Phi, \Lambda) = \frac{1}{Z} \exp\left\{ -\sum_{\alpha=1}^{k} < \lambda^{(\alpha)}, H^{(\alpha)}(\Gamma) > \right\}. \tag{9}$$

The Lagrange multipliers $\lambda^{(\alpha)}, \alpha = 1, 2, \ldots, k$ are solved from the constraint equations or, equivalently, by maximum likelihood estimation. They can be computed by the following iterative equations:

$$\frac{d\lambda^{(\alpha)}}{dt} = E_{p(\Gamma;\Phi,\Lambda)}\left[H^{(\alpha)}(\Gamma)\right] - \mu_{\text{obs}}^{(\alpha)}, \quad \alpha = 1, 2, \ldots, k, \tag{10}$$

where t is time step, and $E_{p(\Gamma;\Phi,\Lambda)}[H^{(\alpha)}(\Gamma)]$ is the expected histogram with respect to the current model $p(\Gamma;\Phi,\Lambda)$. When the above dynamical equation converges, i.e., $\frac{d\lambda^{(\alpha)}}{dt} = 0$, then $p(\Gamma;\Phi,\Lambda)$ duplicates the observed statistics. It is well-known that Λ has a unique solution as $\log p(\Gamma;\Phi,\Lambda)$ is straight concave with respect to Λ [32], provided that the constraint equations are consistent.

In (10), for a given Λ, $E_{p(\Gamma;\Phi,\Lambda)}[H^{(\alpha)}(\Gamma)]$ is hard to compute analytically. In general, one needs to simulate a Monte Carlo Markov Chain(MCMC) walking randomly in the shape Ω_Γ. When this MCMC becomes stationary, it samples the distribution $p(\Gamma;\Phi,\Lambda)$ and we shall discuss the design of MCMC in the next section. Given Φ and Λ, we sample a set of shapes $\Gamma_j^{\text{syn}}, j = 1, 2, \ldots, M$ from $p(\Gamma;\Phi,\Lambda)$ and we estimate $E_{p(\Gamma;\Phi,\Lambda)}[H^{(\alpha)}]$ by the sample mean $\mu_{\text{syn}}^{(\alpha)}$ computed in the same way as for $\mu_{\text{obs}}^{(\alpha)}$ in (4).

So, Λ is updated by

$$\frac{d\lambda^{(\alpha)}}{dt} = \mu_{\text{syn}}^{(\alpha)} - \mu_{\text{obs}}^{(\alpha)}, \quad \alpha = 1, 2, \ldots, k. \tag{11}$$

The whole learning process simulates an inhomogeneous Markov and this computational scheme has been successfully applied to texture modeling by Zhu et al. [32].

6.3 The Learned Shape Model

Suppose we adopt the six features as discussed in Section 4, an ME model is

$$p(\Gamma) = \frac{1}{Z}\exp\Big\{-\int \lambda^{(1)}(\kappa(s)) + \lambda^{(2)}(\nabla\kappa(s)) + \lambda^{(3)}(r(s)) + \lambda^{(4)}(\nabla r(s)) + \lambda^{(5)}(\nabla^2 r(s))ds + \gamma\|B\|\Big\}. \tag{12}$$

In (12), $\|B\|$ is the number of discontinuities of $\psi(s)$. The number of branches of a shape is $\|B\|/4$. The model in (12) has some desirable properties.

1. Multiple features and Gestalt laws—both region-based and contour-based—are fused into a single probability measure on Markov random fields.
2. Shape features are weighted by the learned potential functions and correlations between features are taken into account in the learning process.
3. With more features selected, $p(\Gamma)$ can be extended to account for the complexity and frequency or likelihood of shapes that occurs in nature. Thus, as $p(\Gamma)$ approaches $f(\Gamma)$, it provides the optimal coding length $-\log p(\Gamma)$ for natural shapes.

7 Design MCMC for Shape Sampling

In this section, we discuss a Markov Chain Monte Carlo method for sampling $p(\Gamma;\Phi,\Lambda)$ and we also demonstrate experiments on sampling uniform distributions in the shape space Ω_Γ.

Designing a sampling process is important for the following reasons. First, it is a necessary step for estimating the potential functions Λ. Second, it provides a natural way for verifying the learned models. If a shape model is close to the underlying truth, then the typical set of samples from this model should be similar to the observed shapes judged by human perception. Third, it is also an engine for shape inference from real images when the learned shape model is used as a prior distribution in vision tasks.

In Section 3, a shape

$$\Gamma = ((x_0, y_0), (x_1, y_1), \ldots, (x_{N-1}, y_{N-1}))$$

is defined on a continuous space Ω_Γ. In computer implementation, the cruel reality is that the nodes $(x_i, y_i), = 0, 1, 2, \ldots$ can only be described in finite precision and we denote by τ the unit length of the lattice.[6] The sampling process starts with a simple closed curve, such as a circle or rectangle with N nodes. At each step, it randomly picks up a node (x_i, y_i), and proposes to move it to (x_i', y_i')—a small perturbation moving Γ to Γ'. If (x_i', y_i') violates the two hard constraints, then the proposal is rejected; otherwise, it is accepted with a probability computed from the model $p(\Gamma;\Phi,\Lambda)$. This is known as a Metropolis-Hastings algorithm and it simulates a stochastic process—random walk in the shape space Ω_Γ. If it runs long enough, this stochastic process converges to its equilibrium. When this occurs, its status Γ is subject to distribution $p(\Gamma;\Phi,\Lambda)$ regardless of its starting status and Γ is a synthesized shape from $p(\Gamma;\Phi,\Lambda)$.

There are two immediate questions that we have to address before we discuss the algorithm in detail. First, since, in shape constraint II of Section 3, we have introduced the ϵ because of discretization, what is the effect of ϵ in stochastic sampling? Second, how do we diagnose whether or not the random walk becomes stationary? There is no precise answer to the first question due to the complexity of the shape space Ω_Γ. If $\frac{\epsilon}{ds} \to 0$, then we expect the ϵ effect should vanish; however, if ϵ is too small, the sampling process converges very slowly. We argue that the choice of the ratio $\frac{\epsilon}{ds}$ should also depend on the precision of human perception. We did two comparison experiments. In one experiment, we used $\epsilon = 2\tau$ and $ds = 10\tau$ and, in the other, we used $\epsilon = 2\tau$, and $ds = 20\tau$. There were no noticeable differences for the learning process except that the convergence becomes slower for the latter. There is no good answer for the second question either; the way we use to diagnose the convergence is to monitor the convergence of the marginal distributions $\mu_{\text{syn}}^{(\alpha)}, \alpha = 1, 2, \ldots$ estimated from the most recent samples and we monitor the sampled shapes by our perception. The objective of the sampling process is to match $\mu_{\text{syn}}^{(\alpha)}$ to $\mu_{\text{obs}}^{(\alpha)}$, and our experiments in later sections show that the MCMC process can achieve this goal.

Now, we discuss the details of the sampling algorithm.

Suppose, at a certain step, a node $A = (x_i, y_i)$ is chosen at random and A is allowed to move in a local neighborhood $\{x_i - \tau, x_i, x_i + \tau\} \times \{y_i - \tau, y_i, y_i + \tau\}$—the solid square in Fig. 7a. Suppose there are N_A $(1 \le N_A \le 9)$ valid moves out of the nine positions because putting node A in any of the other $(9 - N_A)$ positions may violate the two hard shape constraints. Now, we **propose** moving A to one of the N_A positions B at equal chance $K(A \to B) = \frac{1}{N_A}$. Similarly, we

6. One can represent the nodes with subpixel accuracy by setting $\tau = 1/5$ or $1/10$ of a pixel width.

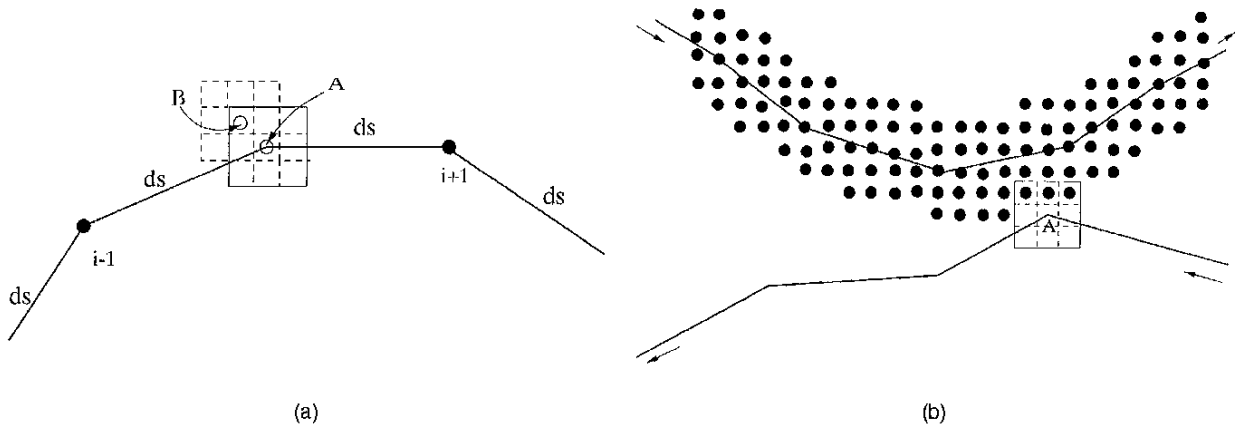


Fig. 7. (a) The proposal for moving a node from position A to B. (b) The firewall preventing the curve from self-intersecting itself.

compute N_B as the number of valid moves in the neighborhood of B—the dashed square. Obviously, it is valid to move from B back to A and the chance for proposing a move from B to A is $K(B \rightarrow A) = \frac{1}{N_B}$. This guarantees that the sampling process is reversible—an important precondition for designing the stochastic process.

The move from A to B is one step of random walk and we denote by Γ_A and Γ_B the two curves, respectively. The proposal is accepted with probability

$$\alpha(A \rightarrow B) = \min\left(\frac{K(B \rightarrow A)p(\Gamma_B; \Phi, \Lambda)}{K(A \rightarrow B)p(\Gamma_A; \Phi, \Lambda)}, 1\right).$$

In the above equation, $p(\Gamma_A; \Phi, \Lambda)$ and $p(\Gamma_B; \Phi, \Lambda)$ are, due to Markov property, computed as $p((x_i, y_i)| \cup_{j \in \partial i} \{(x_j, y_j)\})$, where ∂i is the local neighborhood set of node i. For example, if $\phi^{(1)} = \kappa(s)$ is the only feature chosen in the model, then the computation of the probability of point (x_i, y_i) will only involve (x_{i-1}, y_{i-1}) and (x_{i+1}, y_{i+1}). If region-based properties are integrated in the model, then we compute $r(s)$ from the current mapping function $\psi(s)$ for $p(\Gamma_A; \Phi, \Lambda)$. The mapping function ψ for Γ_B should be updated locally[7] and $r(s)$ at node i is recomputed for estimating $p(\Gamma_B; \Phi, \Lambda)$.

In summary, the transpose probability is

$$P(A \rightarrow B) = K(A \rightarrow B)\alpha(A \rightarrow B)$$

and the random walk satisfies the detailed balance equation

$$p(\Gamma_A; \Phi, \Lambda)P(A \rightarrow B) = p(\Gamma_B; \Phi, \Lambda)P(B \rightarrow A).$$

This guarantees that if the stochastic process walks long enough, its statuses, i.e., Γ, are samples from $p(\Gamma; \Phi, \Lambda)$.

The algorithm for sampling $p(\Gamma; \Phi, \Lambda)$ is given as follows:

Algorithm I: stochastic algorithm for shape sampling

Step 1: initialize $\Gamma = ((x_0, y_0), \ldots, (x_{N-1}, y_{N-1}))$.
Step 2: initialize the mapping function $\psi(s)$ for Γ.
if region-based features are chosen in Φ.
Step 3: sweep $\leftarrow$ 0.
Step 4: for $count = 1$ to N do.
Step 5: Pick up $i \in [0, N-1]$ at random, node i is at position A.
Step 6: Compute N_A, pick up a position B at random, compute N_B.
Step 7: Update ψ for the new curve, if necessary.
Step 8: Compute $\alpha(A \rightarrow B)$.
Step 9: Draw random number $r \in [0, 1)$ at uniform dist.
Step 10: If $r < \alpha$, then move node i from A to B.
Step 11: sweep $\leftarrow$ sweep+1
Step 12: stop, if sweep $>$ threshold, or go to step 4.

In our experiment, to prevent a curve from intersecting itself, we construct a "firewall" of width 5τ along the curve (see the black dots in Fig. 7b). The cells along the line segment from node i to node $i+1$, displayed by the dots in Fig. 7b, are labeled as i, and a node j is prevented from moving in the firewall cells labeled i if $(|j - i| \, mod \, N) \geq 2$. For example, node A in Fig. 7 cannot move in the three dotted cells and $N_A = 6$.

Our first experiment is to sample a uniform distribution on Ω_Γ. One hundred random shapes are recorded after 1.5 million sweeps with $N = 200$, $ds = 10\tau, \epsilon = 2\tau$. Fig. 8 displays four of the sampled shapes. To study how the statistics of these random shape vary with resolutions, we subsampled the 100 shapes once and twice and obtained shapes of 100 nodes and 50 nodes, respectively. Fig. 9a shows the average histograms of 100 synthesized shapes at three scales: $H^{(m)}_{200}, m = 0, 1, 2$ by dash-dotted, dashed, and solid curve, respectively. m is the times of subsampling. Obviously, the histograms change over scales. With a higher resolution, local structures of the random shapes become richer and display fractal properties. This scale-sensitivity is in sharp contrast to the pseudo-scale-invariant property of natural shapes in Fig. 5.

Our second experiment is to diagnose the convergence of the Markov chain, and to study the effect of the resolution N in the synthesized shapes. We sample a second group of 100 shapes with $N = 400, ds = 10\tau, \epsilon = 2\tau$, and a third group of 100 shapes with $N = 100, ds = 10\tau, \epsilon = 2\tau$. See our technical report for some of these shapes [34].

7. As the mapping function $\psi(s)$ is computed in a Markov random field, this updating $\psi(s)$ only involves local computations [33].

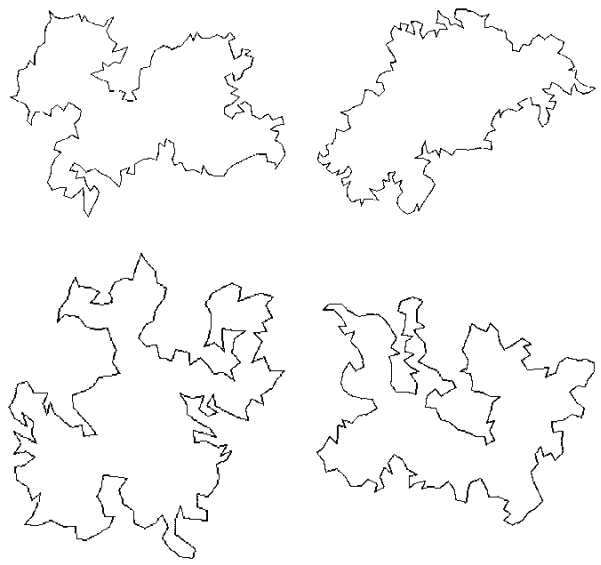


Fig. 8. Four of the sampled shapes recorded at different time steps of a stochastic process designed for uniform distribution. $N = 200$, $ds = 10\tau$, $\epsilon = 2\tau$.

Fig. 9b shows two groups of histograms $H_N^{(m)}$, where the subscript N is the number of nodes used in the sampling processes and the superscript (m) denotes the scales. The first group includes two broad histograms (all have 200 nodes): $H_{200}^{(0)}$ (dashed) and $H_{400}^{(1)}$ (solid). The second group includes three sharper curves (all have 100 nodes): $H_{100}^{(0)}$ (dash-dotted), $H_{200}^{(1)}$ (dashed), and $H_{400}^{(2)}$ (solid). It is clear that the histograms observed in the same resolution are very close, which indicates the consistency of the sampling processes with $N = 100$, 200, and 400. Perceptually, the random shapes with $N = 200$ nodes have no noticeable differences from the shapes subsampled from the random shapes with $N = 400$ nodes (see our technical report for some of these shapes [34]).

8 Feature Pursuit by Maximizing Nonaccidental Statistics

In this section, we discuss how to pursue important features for shape modeling.

We start with $\Phi_0 = \emptyset$ and, thus, $p(\Gamma)$, a uniform distribution. At each step k, given $\Phi_k = \{\phi^{(\alpha)}, \alpha = 1, 2, \ldots, k.\}$ and a model $p(\Gamma; \Phi_k, \Lambda_k)$ is learned according to Algorithm I. As a result, we obtain a set of synthesized shapes $\{\Gamma_i^{\text{syn}}, i = 1, 2, \ldots, M'\}$ and $\mu_{\text{syn}}^{(\alpha)} = \mu_{\text{obs}}^{(\alpha)}$, for $\alpha = 1, 2 \ldots, k$. Now, for any new feature $\phi^{(\beta)} \notin \Phi_k$, we compute the histograms $\mu_{\text{syn}}^{(\beta)}$ and $\mu_{\text{obs}}^{(\beta)}$ from the synthesized and the observed shapes, respectively. We notice that $\mu_{\text{syn}}^{(\beta)}$ is the *accidental statistic* for feature $\phi^{(\beta)}$ and it accounts for the correlation between $\phi^{(\beta)}$ and the chosen features in Φ_k.

Definition. *Given Φ_k and $p(\Gamma; \Phi_k, \Lambda_k)$, the* ***nonaccidental statistic*** *for feature $\phi^{(\beta)}$ is the distance between $\mu_{\text{obs}}^{(\beta)}$ and $\mu_{\text{syn}}^{(\beta)}$.*

At step $k+1$, we should choose the feature which has the largest nonaccidental statistic. The distance between $\mu_{\text{obs}}^{(\beta)}$ and $\mu_{\text{syn}}^{(\beta)}$ could be measured by an L_1 norm or a quadratic form—the Mahanalobis distance.

In modeling texture [32], Zhu et al. proposed a minimum entropy principle for selecting the optimal set Φ of features from a dictionary of Gabor filters. It is proven that each feature selection step from $p(\Gamma; \Phi_k, \Lambda_k)$ to $p(\Gamma; \Phi_{k+1}, \Lambda_{k+1})$ is a steepest descent way for minimizing the Kullback-Leibler distance $D(f||p)$ between the true distribution $f(\Gamma)$ and the model $p(\Gamma)$. $D(f||p)$ is a conventional measure for the goodness of the learned model $p(\Gamma)$. For shape modeling,

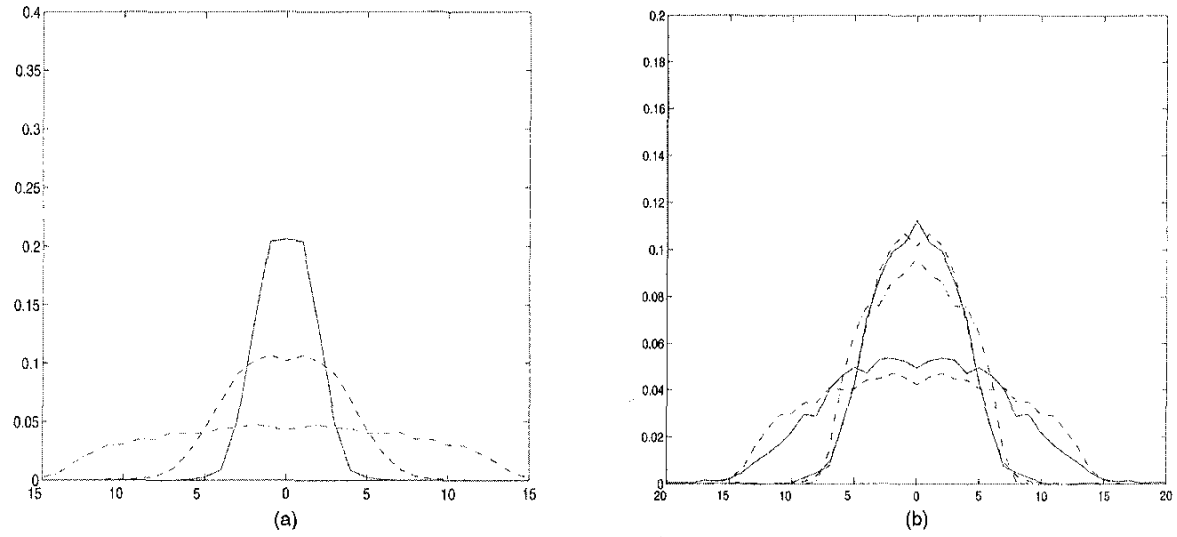


Fig. 9. (a) Histograms of $\kappa(s)$ averaged over 100 samples at three scales $H_{200}^{(0)}$ (dash-doted), $H_{200}^{(1)}$ (dashed), and $H_{200}^{(2)}$ (solid), respectively. (b) The two broad histograms of $\kappa(s)$ are $H_{200}^{(0)}$ (dashed) and $H_{400}^{(1)}$ (solid), respectively, and the three sharper histograms are $H_{100}^{(0)}$ (dash-dotted), $H_{200}^{(1)}$ (dashed), and $H_{100}^{(2)}$ (solid), respectively.

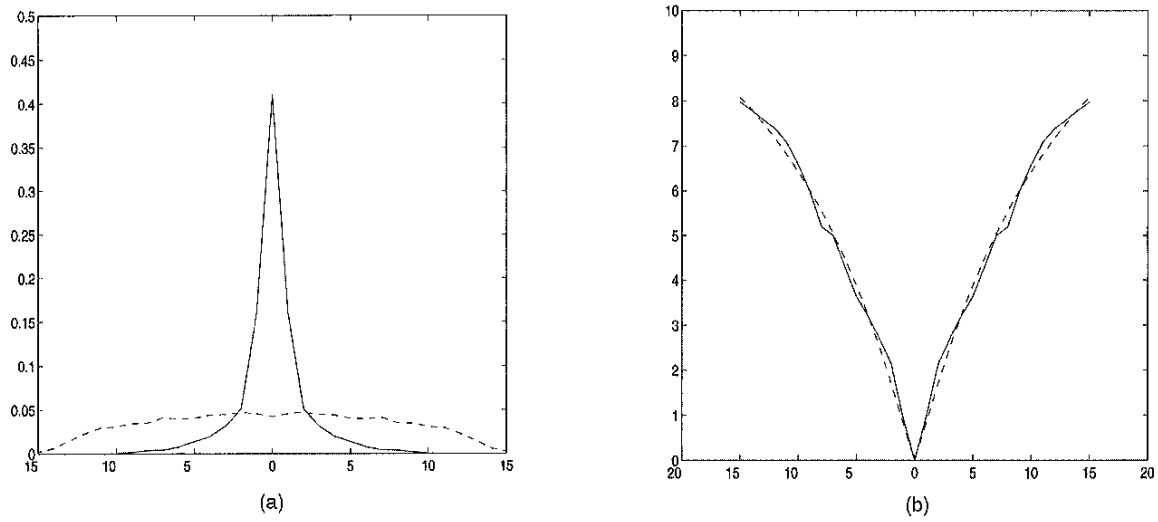


Fig. 10. (a) $\mu_{\text{obs}}^{(1)}$ (solid) versus $\mu_{\text{syn}}^{(1)}$ of a uniform distribution for $\phi^{(1)}(s)$. These two curves appeared in Fig. 5a and Fig. 9a, respectively. (b) The learned $\lambda^{(1)}()$ for $\phi^{(1)}(s) = \kappa(s)$ in a nonparametric form (solid curve) is fit to $\eta(z) = a(1 - 1/(1 + (z/b)^{\gamma}))$ with $a = 15$, $b = 13$, $\gamma = 1.1$; $dz = \frac{\pi}{16 \times 200}$ is the unit length of the horizontal axis $\kappa(s)$.

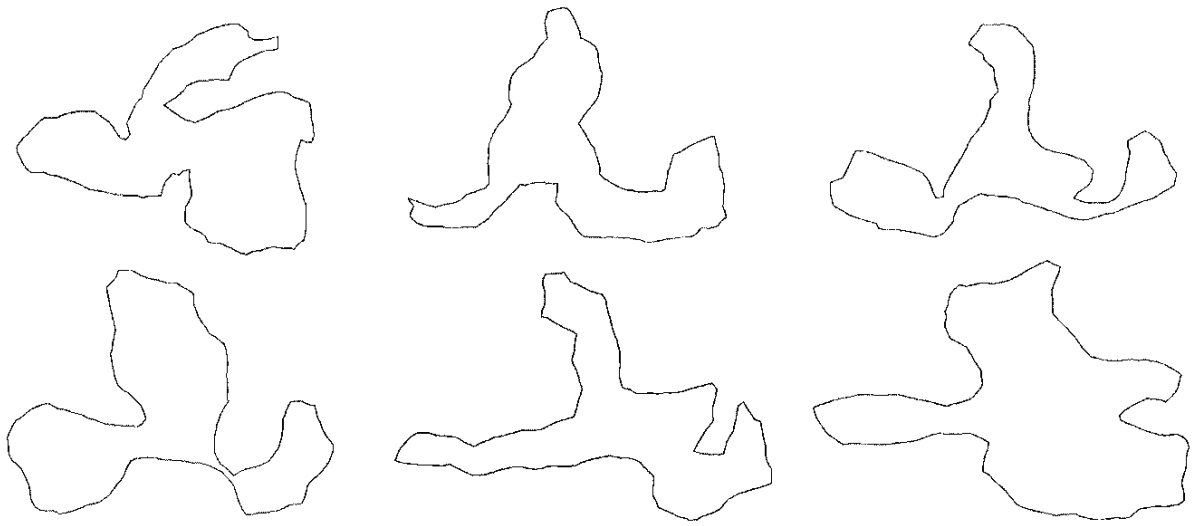


Fig. 11. Six of the synthesized shapes with curvature histogram matched to animate shapes, $\mu_{\text{syn}}^{(1)} = \mu_{\text{obs}}^{(1)}$. The histograms of these synthesized shapes are shown by the dashed curves in Fig. 12.

the Gestalt laws play the same role as Gabor filters for texture modeling; however, there are far fewer Gestalt laws than Gabor filters. So, we choose not to discuss the feature selection issue explicitly and the proof for the following proposition is referred to our texture paper [32].

Proposition *At each step, choosing a feature which has the maximum nonaccidental statistic is a steepest descent step of minimizing the Kullback-Leibler distance* $D(f \,||\, p)$.

In summary, the overall algorithm for shape learning is listed below.

Algorithm II: algorithm for shape learning

Step 1: given $\{\Gamma_i^{\text{obs}}$, for $i = 1, 2, \ldots M\}$, compute $\mu_{\text{obs}}^{(\alpha)}, \alpha = 1, 2, \ldots$

Step 2: $k = 0$, initialize $\Phi_0 = \emptyset, \Lambda_0 = 0$, and $p(\Gamma; \Phi_0, \Lambda_0)$ a uniform dist.

Step 3: sample $\Gamma_i^{\text{syn}}, i = 1, 2 \ldots, M'$ from $p(\Gamma; \Phi, \Lambda)$ using Algorithm I.

Step 4: compute $\mu_{\text{syn}}^{(\beta)}$ for candidate features $\phi^{(\beta)}$.

Step 5: adding a new feature, $\Phi \leftarrow \Phi \cup \{\phi^{(k)}\}$, $\lambda^{(k)} \leftarrow 0$, $k \leftarrow k + 1$.

Step 6: $\lambda^{(\alpha)} \leftarrow \lambda^{(\alpha)} + \Delta t(\mu_{\text{syn}}^{(\alpha)} - \mu_{\text{obs}}^{(\alpha)})$, $\alpha = 1, 2 \ldots, k$.

Step 7: sample $p(\Gamma; \Phi, \Lambda)$ for $10,000$ sweeps using Algorithm I.

Step 8: compute $\mu_{\text{syn}}^{(\alpha)}, \alpha = 1, 2 \ldots, k$ from a set of recently sampled shapes.

Step 9: goto step 6, unless $\|\mu_{\text{syn}}^{(\alpha)} - \mu_{\text{obs}}^{(\alpha)}\| < \rho$ for $\alpha = 1, 2, \ldots, k$.

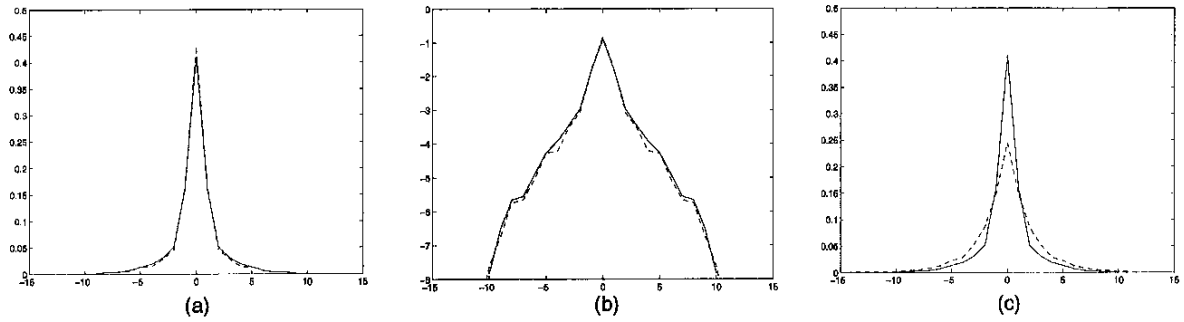


Fig. 12. (a) $\mu_{\text{syn}}^{(1)}$ (dashed) vs. $\mu_{\text{obs}}^{(1)}$ (solid). (b) $\log \mu_{\text{syn}}^{(1)}$ (dashed) vs. $\log \mu_{\text{obs}}^{(1)}$ (solid). (c) $\mu_{\text{syn}}^{(2)}$ (dashed) vs. $\mu_{\text{obs}}^{(2)}$ (solid).

Step 10: goto step 5, unless $\|\mu_{\text{obs}}^{(\beta)} - \mu_{syn}^{(\beta)}\| < \Upsilon$ for all remaining features $\phi^{(\beta)}$.

The whole learning process is computationally very intensive, especially when the region-based features are computed.

9 Experiments

In this section, we demonstrate experiments on shape learning, following algorithms I and II.

9.1 Experiment I

We start with $\Phi = \emptyset$ and $p(\Gamma; \Phi, \Lambda)$ a uniform distribution. In Fig. 10a, we compare $\mu_{\text{syn}}^{(1)}$(dashed) of the uniform distribution against $\mu_{\text{obs}}^{(1)}$(solid). $\mu_{\text{obs}}^{(1)}$ has a much sharper peak than $\mu_{\text{syn}}^{(1)}$ near zero, $\mu_{\text{obs}}^{(1)}$ measures quantitatively how much colinearity natural shapes have in comparison with $\mu_{\text{syn}}^{(1)}$, the colinearity resulting from **accidental arrangement** in the uniform shapes while the hard constraints are taken into account. We found that $\|\mu_{\text{syn}}^{(1)} - \mu_{\text{obs}}^{(1)}\| > \|\mu_{\text{syn}}^{(2)} - \mu_{\text{obs}}^{(2)}\|$, which means $\kappa(s)$ is a more significant property than $\nabla\kappa(s)$—the cocircularity in natural shapes. In this initial step, we don't compare other region-based features as it is quite unreliable to compute the symmetric mapping function $\psi(s)$ for the jagged synthesized shapes.

Then, a Gibbs distribution is learned which could reproduce the marginal distribution for feature $\phi^{(1)} = \kappa(s)$

$$p(\Gamma; \Phi_1, \Lambda_1) = \frac{1}{Z} \exp\left\{ - \int \lambda^{(1)}(\kappa(s)) ds \right\}.$$

We adopt the continuous notation for consistency and the learned potential function $\lambda^{(1)}(z)$ is plotted in Fig. 10b. $\lambda^{(1)}(z)$ is close to $|z|$ near zero, but it has flat tails to preserve large curvatures. Fig. 11 displays six of the sampled shapes from the learned model. From the set of sampled shapes, we compute $\mu_{\text{syn}}^{(\alpha)}, \alpha = 1, 2$. Fig. 12a displays $\mu_{\text{syn}}^{(1)}$ (dashed)—a marginal distribution of $p(\Gamma; \Phi_1, \Lambda_1)$ against $\mu_{\text{obs}}^{(1)}$ (solid), and Fig. 12b plots $\log \mu_{\text{syn}}^{(1)}$ (dashed) against $\log \mu_{\text{obs}}^{(1)}$ (solid). Obviously, the shape model $p(\Gamma; \Phi_1, \Lambda_1)$ reproduces the observed curvature histogram precisely. $\mu_{\text{syn}}^{(2)}$ is plotted

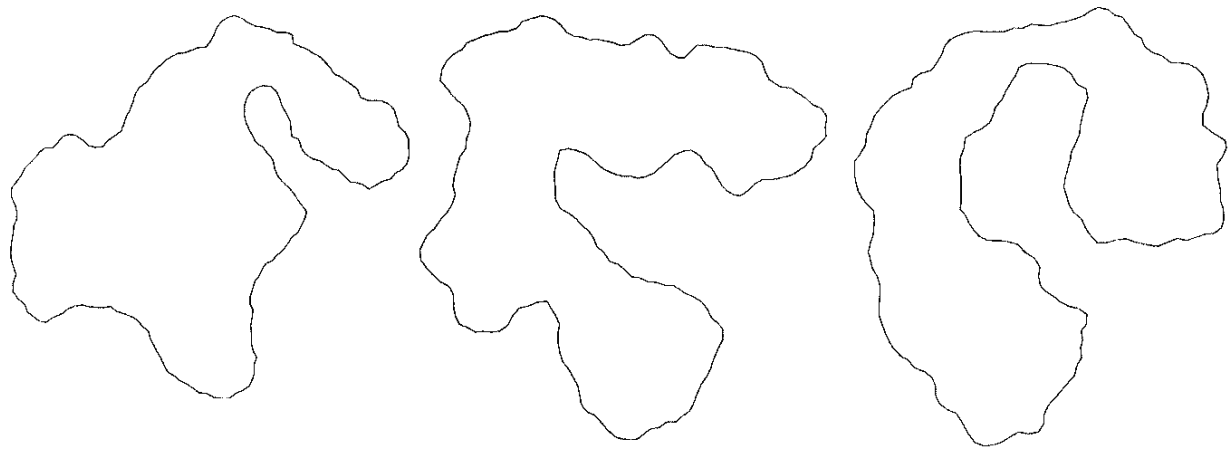


Fig. 13. Sampled shapes from a Gaussian model after 2 million sweeps. The variance of $\kappa(s)$ for the animal shape is 0.1665; the sampled shapes have variance 0.1657, which is a very close match.

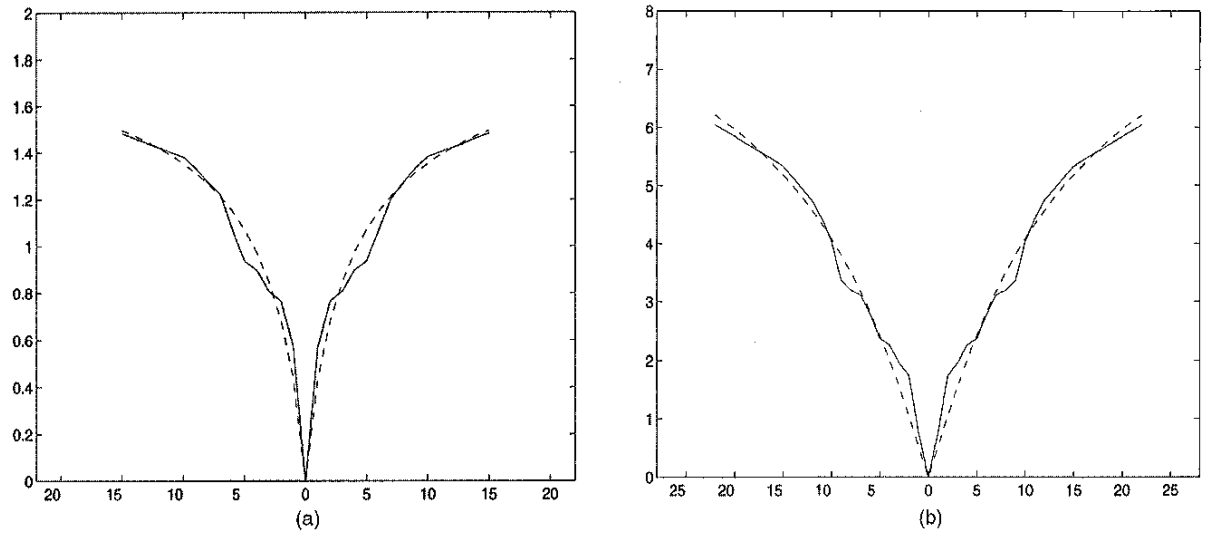


Fig. 14. The learned potential functions. (a) $\lambda^{(1)}(z)$ (solid) and the fitting curve $\eta(z) = a(1 - 1/(1 + (z/b)^{\gamma}))$ with $a = 1.95$, $b = 4$, $\gamma = 0.9$. (b) $\lambda^{(2)}(z)$ (solid) and the fitting curve $\eta(z) = a(1 - 1/(1 + (z/b)^{\gamma}))$ with $a = 10$, $b = 14$, $\gamma = 1.1$.

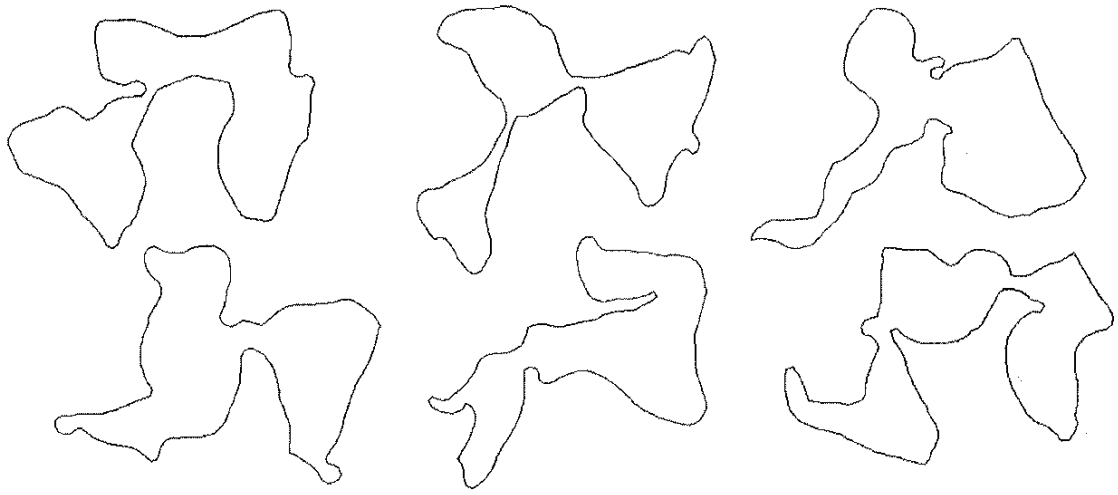


Fig. 15. Six of the synthesized shapes with $\mu^{(\alpha)}_{\text{syn}} = \mu^{(\alpha)}_{\text{obs}}$, $\alpha = 1, 2$. Note that these shapes are smoother than the shapes in Fig. 11.

against $\mu^{(2)}_{\text{obs}}$ in Fig. 12c, so the sampled shapes doesn't have as much cocircularity as in the observed shapes.

For comparison, we also sampled a Gaussian distribution

$$p(\Gamma) = \frac{1}{Z}\exp\left\{-\int \kappa^2(s)/\sigma^2 ds\right\}$$

with σ chosen to reproduce the same mean and the same variance of the curvature as in the observed shapes. We show three sampled shapes in Fig. 13. These shapes have more jagged boundaries and fewer structures than the shapes in Fig. 11.

9.2 Experiment II

In the second experiment, we choose $\Phi_2 = \{\kappa(s), \nabla\kappa(s)\}$, and the model is learned to match both $\mu^{(1)}_{\text{obs}}$ and $\mu^{(2)}_{\text{obs}}$

$$p(\Gamma; \Phi_2, \Lambda_2) = \frac{1}{Z}\exp\left\{-\int \lambda^{(1)}(\kappa(s)) + \lambda^{(2)}(\nabla\kappa(s))ds\right\}.$$

The learned $\lambda^{(1)}(z)$ and $\lambda^{(2)}(z)$ are shown as the solid curves in Fig. 14a and Fig. 14b, respectively. Six of the sampled shapes are shown in Fig. 15. The average histograms $\mu^{(\alpha)}_{\text{syn}}, \alpha = 1, 2$ for 45 synthesized shapes are shown in Fig. 16a and Fig. 16b by the dashed curves and they match $\mu^{(\alpha)}_{\text{obs}}, \alpha = 1, 2$ closely.

The shapes in Fig. 15 share the same amount of colinearity and cocircularity as the observed animate shapes. These shapes are smooth and we also notice that long circular arcs are formed through the propagations of local interactions in Markov random fields.

We want to emphasize two issues in this experiment:

- The synthesized shapes have pseudoscale invariant property like the observed shapes. For example, we

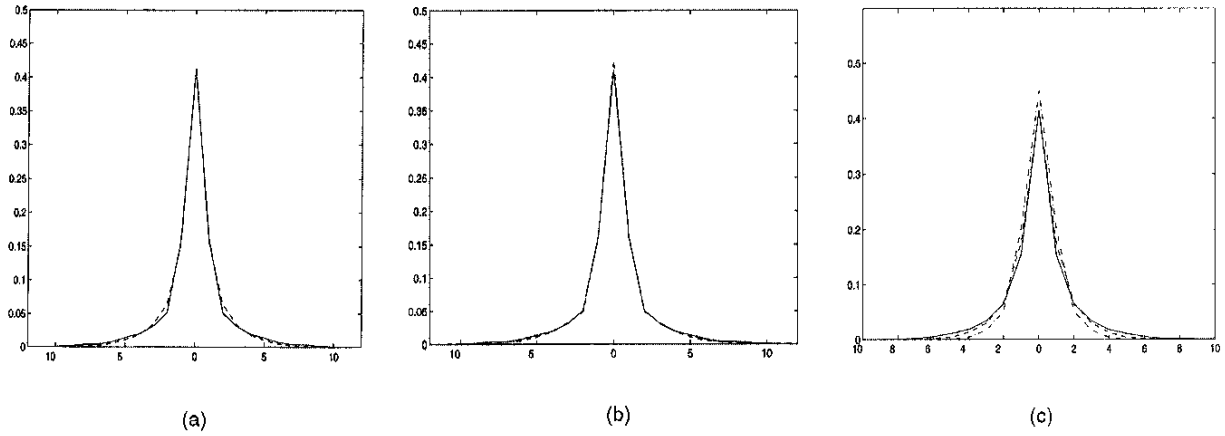


Fig. 16. (a) the histogram of $\kappa(s)$ averaged over 45 synthesized shapes (dashed curve) and $\mu_{\text{obs}}^{(1)}$ solid curve. (b) the histogram of $\nabla\kappa(s)$ averaged over 45 synthesized shapes (dashed curve) and $\mu_{\text{obs}}^{(2)}$ (solid curve). (c) The histograms of $\kappa(s)$ averaged over 45 synthesized shapes at scale 0 (solid curve), scale 1 (dashed curve), and scale 2 (dash-dotted curve).

collect 45 sampled shapes and subsample them once and twice and generate two new sets of shapes at scale 1 and scale 2. Fig. 16c shows the average curvature histograms for the three scales. This indicates that *the choice of N is not critical in shape modeling as long as it is large enough to approximate the curve up to the precision of human perception.*

- The learned potential function $\lambda^{(1)}$ in Fig. 14a is different from that in Fig. 10b. This change reflects the correlation between the two features. This demonstrates that the learning process could account for interdependency between chosen features.

9.3 Experiment III

Our third experiment incorporates the region-based features into the shape model.

Although the shapes in Fig. 15 reproduce the exact amount of colinearity and cocircularity, they are very blob-like and elongated parts, like limbs of animals, are missing. The lack of region-based features are reflected in the histograms plotted in Fig. 17. We compute $\mu_{syn}^{(\alpha)}, \alpha = 3, 4, 5$ for features $\phi^{(3)}(s) = r(s)$, $\phi^{(4)}(s) = \nabla r(s)$, and $\phi^{(5)}(s) = \Delta r(s)$ from a set of 45 synthesized shapes of $p(\Gamma; \Phi_2, \Lambda_2)$. $\mu_{syn}^{(\alpha)}, \alpha = 3, 4, 5$ are plotted as dashed curves in Fig. 17a, Fig. 17b, Fig. 17c, respectively. In contrast, $\mu_{\text{obs}}^{(\alpha)}, \alpha = 3, 4, 5$ are the solid curves. Again, the differences measure the nonaccidental statistics in natural shapes.

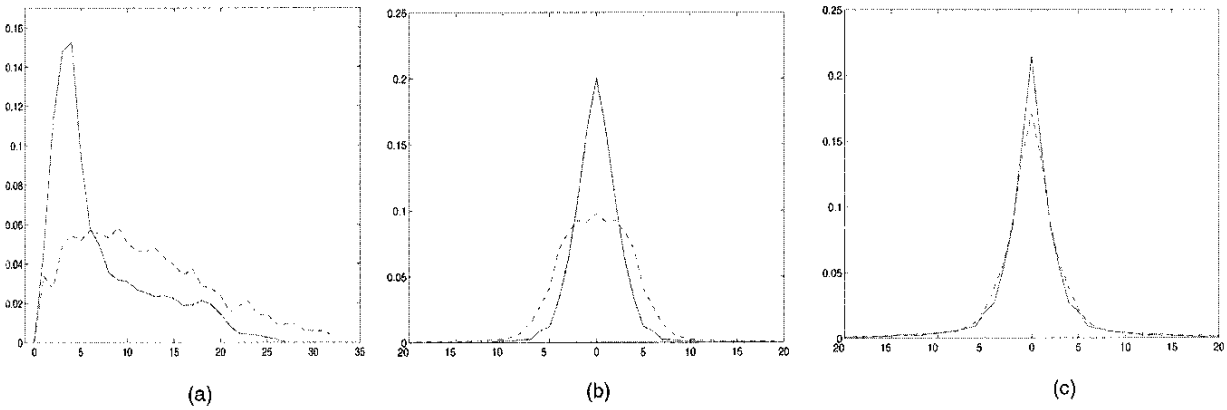


fig. 17. Average histograms of shapes sampled from $p(\Gamma; \Phi_2, \Lambda_2)$. $\mu_{\text{syn}}^{(\alpha)}$ (dashed) vs. $\mu_{\text{obs}}^{(\alpha)}$ (solid), (a) $\alpha = 3$, (b) $\alpha = 4$, (c) $\alpha = 5$.

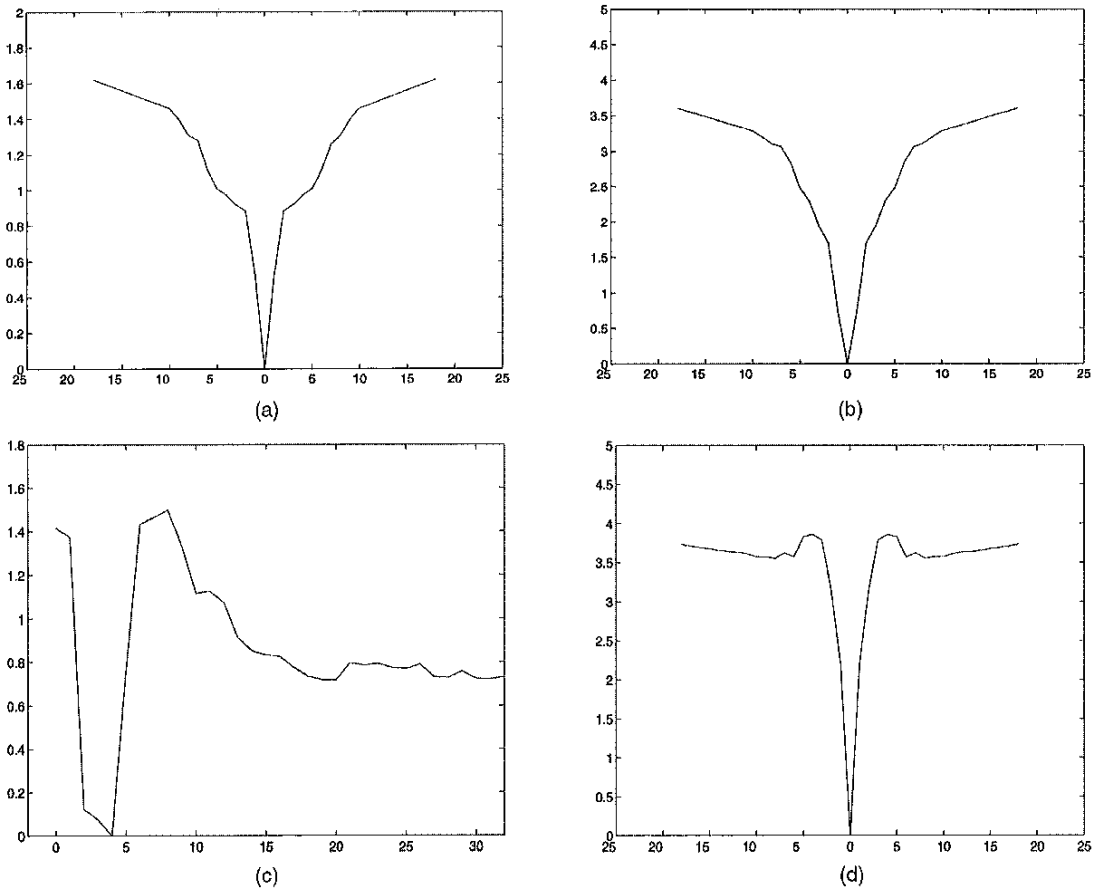


Fig. 18. The learned potentional functions in $p_3(\Gamma; \Phi_3, \Lambda_3)$. (a) $\lambda^{(1)}(z)$. (b) $\lambda^{(2)}(z)$. (c) $\lambda_{(3)}(z)$. (d) $\lambda^{(4)}(z)$.

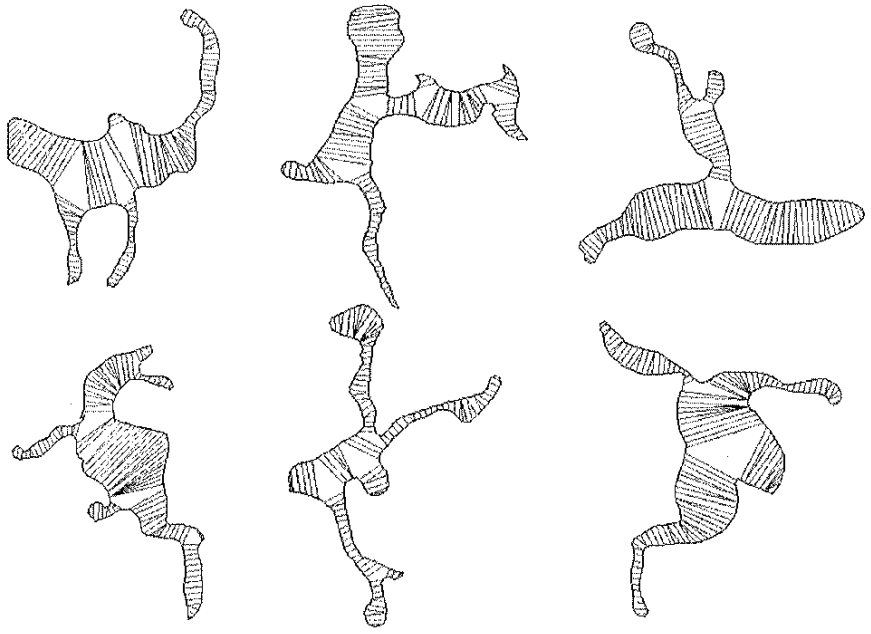


Fig. 19. Six of the synthesized shapes from $p_4(\Gamma; \Phi_4, \Lambda_4)$.

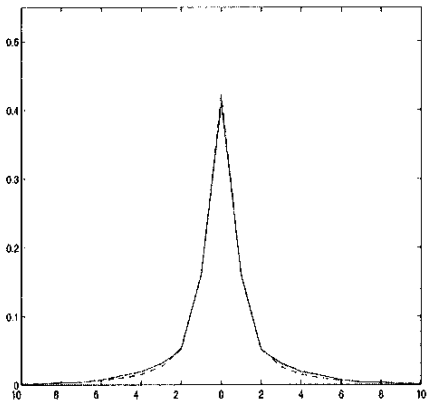


Fig. 20. The solid line is the average histogram of curvature $\kappa(s)$ for 22 animals (flipped to 44). The dashed line is the average histogram of $\kappa(s)$ for 1,100 fish—a different dataset.

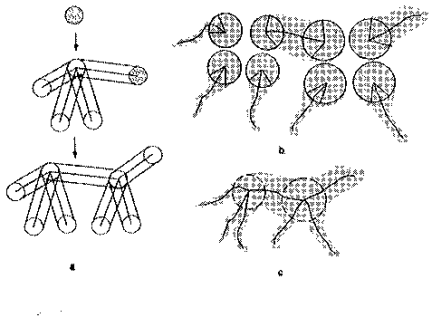


Fig. 21. The general deformable model for describing a dog in the FORMS system (Zhu and Yuille 1996). (a) The grammar for generating the graph structure of an animal. (b) The deformation of parts. (c) Composing the shape through the hinge joints.

As the difference between $\mu^{(5)}_{\text{syn}}$ and $\mu^{(5)}_{\text{obs}}$ is small, we choose two new features $r(s), \nabla r(s)$ and learn a new model

$$p_4(\Gamma; \Phi_4, \Lambda_4) = \frac{1}{Z}\exp\Big\{-\int \lambda^{(1)}(\kappa(s)) + \lambda^{(2)}(\nabla\kappa(s)) + \lambda^{(3)}(r(s)) + \lambda^{(4)}(\nabla r(s))ds\Big\}.$$

The learned $\lambda^{(\alpha)}, \alpha = 1, 2, 3, 4$ are shown in Fig. 18. Six of the sampled shapes are displayed in Fig. 19 together with their mapping functions $\psi(s)$.

This experiment demonstrates that the region-based features are crucial for shape modeling. Although none of these synthesized shapes are identifiable as specific animate objects in our environments, the sampled shapes resemble parts of real world objects, exactly as we expected for middle level vision. We shall debate this issue in the next section.

10 Discussion

In this paper, a theory for shape modeling and learning is proposed based on the maximum entropy principle and various Gestalt laws are embedded into the Markov random field models. The paper also provides quantitative measures for the nonaccidental arrangement by distances between the observed statistics and the statistics of random shapes. The differences between $\mu^{(\alpha)}_{\text{syn}}$ and $\mu^{(\alpha)}_{\text{obs}}$ are evidence for the ecological reasons underlying the Gestalt laws —colinearity, cocircularity, proximity, parallelism, and symmetry—identified in experiments of human visual perception. Thus, the article provides a firm mathematical justification for some Gestalt laws.

The learned models are limited in a few aspects.

First, they are biased by the training shapes that we selected. We have studied a different database of shapes—1,100 fish contours, and the statistics in this dataset are found to be very similar to those of our training set (see Fig. 20). It is not very surprising because both datasets are animate objects and histogram is averaged over large samples (the total number of points in the dataset is over 5,000). However, the statistics are likely to change for the ensemble of polygon shapes such as rectangles and triangles, whose curvature histograms should have sharper peak at zero.

Second, the neighborhood structures of our models are limited by the Gestalt laws. There are very few choices for generic shape features. The Gestalt laws are known to be the most important generic features for human perception, if they are not the best. For the same reason, we don't discuss the histogram variations between animate shapes and we have assumed that $\mu^{(\alpha)}_{\text{obs}}(z) = \mu^{(\alpha)}(z)$ in this paper. Indeed, studying the histogram variation and the estimation error of the observed histograms is important to decide when to stop choosing a feature; this was discussed in our texture modeling paper [32].

Third, the models do not account for high level shape properties. For example, some portions of the sampled shapes in Fig. 19 may resemble parts of animals, such as legs, tails, and heads, but these parts are not assembled in a proper way.

We now discuss how the models can be extended to overcome the third limitation. One can easily compute the medial axes (or skeletons) for the shapes in Fig. 19 by connecting the centers of the mapping line segments and a detailed algorithm is referred to in a companion paper [33]. The skeleton forms a graph representation of shape. For example, Fig. 21 shows a shape model for high level recognition in the FORMS system proposed by Zhu and Yuille [30]. The skeleton of a dog is generated by a grammar and the seven deformed parts are joined by two hinge joints. In fact, it is not hard to argue that the deformable models in FORMS are nonhomogeneous maximum entropy models too! This nonhomogeneous model imposes high level structures on the skeleton graphs to model specific objects, e.g., by imposing constraints on the number of branches at each joint and the relative orientations of these branches. In this way, one can sample random shapes for a dog.

We argue that shape models in the middle level should be **compatible** with both low level and high level representations. The Markov random field shape models in this paper can be easily incorporated into the statistical framework for image segmentation [29]. They also provide a mechanism for computing object shapes, as well as their medial axes, from raw images and for passing such a description to the high level.

It is our belief that the random field models may lead to a new way to explain the "field force" in the Gestalt theory. We hope that this paper will stimulate better research results along this direction.

ACKNOWLEDGMENTS

The author is supported by an ARO grant DAAH-04-95-1-0494, U.S. National Science Foundation grant NSF-9877127, and a Microsoft Research gift. The author is grateful to David Mumford and Yingnian Wu for discussions. The author would like to thank two anonymous reviewers for comments that led to improving the presentation of the paper.

REFERENCES

[1] R. Balboa and N.M. Grzywacz, "Power Spectra of Natural Underwater Images and Possible Implications for Vision," preprint, Smith-Kettlewell Eye Research Inst., 1998.
[2] F.L. Bookstein, "Size and Shape Spaces for Landmark Data in Two Dimensions," *Statistical Science*, vol. 1, no. 2, pp. 181-242, 1986.
[3] C.A. Burbeck and S.M. Pizer, "Object Representation by Cores: Identifying and Representing Primitive Spatial Regions," *Vision Research*, vol. 35, no. 13, pp. 1,917-1,930, 1995.
[4] A. Chakraborty, L.H. Straib, and J.S. Duncan, "Deformable Boundary Finding Influenced by Region Homogeneity," *Proc. Computer Vision and Pattern Recognition*, Seattle, Wash., 1994.
[5] S.J. Dickinson, A.P. Pentland, and A. Rosenfeld, "From Volumes to Views: An Approach to 3D Object Recognition," *CVGIP: Image Understanding*, vol. 55, no. 2, pp. 130-154, Mar. 1992.
[6] I.L. Dryden and K.V. Mardia, *Statistical Shape Analysis*. Wiley, 1998.
[7] U. Grenander, *General Pattern Theory*. Oxford Univ. Press, 1993.
[8] U. Grenander, Y. Chow, and K.M. Keenan, *Hands: A Pattern Theoretical Study of Biological Shapes*. New York: Springer-Verlag, 1991.
[9] H. Helmholtz, *Treatise on Physiological Optics*. New York: Dover, 1962 (first published in 1867).
[10] J.E. Hochberg, "Effects of the Gestalt Revolution: The Cornell Symposium on Perception," *Psychological Review*, vol. 64, no. 2, pp. 73-84, 1957.
[11] D.W. Jacobs, *Recognizing 3D Objects Using 2D Images*, unpublished PhD dissertation, Dept. of Electrical Eng. and Computer Science, Massachusetts Inst. of Technology, 1992.
[12] E.T. Jaynes, "Information Theory and Statistical Mechanics," *Physical Review*, vol. 106, pp. 620-630, 1957.
[13] G. Kanizsa, *Organization in Vision*. New York: Praeger, 1979.
[14] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *Proc. Int'l Conf. Computer Vision*, London, 1987.
[15] D.G. Kendall, "A Survey of the Statistical Theory of Shape," *Statistical Science*, vol. 4, no. 2, pp. 87-120, 1989.
[16] D. Keren, D. Cooper, and J. Subrahmonia, "Describing Complicated Objects by Implicit Polynomials," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 1, Jan. 1994.
[17] K. Koffka, *Principles of Gestalt Psychology*. New York: Harcourt, Brace and Company 1935.
[18] I. Kovacs and B. Julesz, "Perceptual Sensitivity Maps within Globally Defined Visual Shapes," *Nature*, vol. 371, pp. 644-646, 1996.
[19] T.S. Lee, D.B. Mumford, S.C. Zhu, and V. Lamme, "The Role of V1 in Shape Representation," *Computational Neuroscience*, K. Bower, ed., New York: Plenum Press, 1997.
[20] M. Leyton, *Symmetry, Causality, Mind*. Cambridge, Mass.: MIT Press, 1992.
[21] L.D. Lowe, *Perceptual Organization and Visual Recognition*. Kluwer Academic, 1985.
[22] D.G. Lowe, "Visual Recognition as Probabilistic Inference from Spatial Relations," *AI and Eye*, A. Blake and T. Troscianko, eds., John Wiley & Sons Ltd., 1990.
[23] K.V. Mardia and I.L. Dryden, "Statistical Analysis of Shape Data," *Biometrika*, vol. 76, pp. 271-281, 1989.
[24] R. Mohan and R. Nevatia, "Perceptual Organization for Scene Segmentation and Description," *IEEE Trans Pattern Analysis and Machine Intelligence*, vol. 14, no. 6, June 1992.
[25] D.B. Mumford, "Elastica and Computer Vision," *Algebraic Geometry and Its Applications*, C.L. Bajaj, ed., New York: Springer-Verlag, 1994.
[26] A.P. Pentland, "Perceptual Organization and the Representation of Natural Form," *Artificial Intelligence*, vol. 28, pp. 293-331, 1986.
[27] L.R. Williams and D.W. Jacobs, "Stochastic Completion Fields: A Neural Model of Illusory Contour Shape and Salience," *Neural Computation*, vol. 9, pp. 837-858, 1997.
[28] A.L. Yuille, "Deformable Templates for Face Recognition," *J. Cognitive Neurosciences*, vol. 3, no. 1, 1991.
[29] S.C. Zhu and A.L. Yuille, "Region Competition: Unifying Snake/Balloon, Region Growing and Bayes/MDL/Energy for Multi-Band Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, Sept. 1996.
[30] S.C. Zhu and A.L. Yuille, "FORMS: A Flexible Object Recognition and Modeling System," *Int'l J. Computer Vision*, vol. 20, no. 3, Dec. 1996
[31] S.C. Zhu and D.B. Mumford, "Prior Learning and Gibbs Reaction-Diffusion," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 11, Nov. 1997.
[32] S.C. Zhu, Y.N. Wu, and D.B. Mumford, "Minimax Entropy Principle and Its Application to Texture Modeling," *Neural Computation*, vol. 9, no. 8, Nov. 1997.
[33] S.C. Zhu, "Stochastic Computation of Medial Axis in Markov Random Field," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1,158-1,169, Nov. 1999.
[34] S.C. Zhu, "Embedding Gestalt Laws in Markov Random Fields," technical report, Dept. of Computer Science, Ohio State Univ., 1998.

Song-Chun Zhu received his BS degree in computer science from the University of Science and Technology of China in 1991. He received his MS and PhD degree in computer science from Harvard University in 1994 and 1996, respectively. He was a research associate in the Division of Applied Math at Brown University during 1996-1997 and he was a lecturer in the Computer Science Department at Stanford University during 1997-1998. He joined the faculty of the Department of Computer and Information Sciences at Ohio State University in 1998, where he is leading the OSU Vision And Learning (OVAL) group. His research is concentrated in the areas of computer and human vision, statistical modeling, and stochastic computing. He has published more than 30 articles on object recognition, image segmentation, texture modeling, visual learning, perceptual organization and performance analysis.