Front. Electr. Electron. Eng. China
DOI 10.1007/s11460-010-0107-x

RESEARCH ARTICLE

Alan YUILLE

An information theory perspective on computational vision

Abstract This paper introduces computer vision from an information theory perspective. We discuss how vision can be thought of as a decoding problem where the goal is to find the most efficient encoding of the visual scene. This requires probabilistic models which are capable of capturing the complexity and ambiguities of natural images. We start by describing classic Markov Random Field (MRF) models of images. We stress the importance of having efficient inference and learning algorithms for these models and emphasize those approaches which use concepts from information theory. Next we introduce more powerful image models that have recently been developed and which are better able to deal with the complexities of natural images. These models use stochastic grammars and hierarchical representations. They are trained using images from increasingly large databases. Finally, we described how techniques from information theory can be used to analyze vision models and measure the effectiveness of different visual cues.

Keywords computer vision, pattern recognition, information theory, minimum description length, Markov random field (MRF) model, stochastic grammars

1 Introduction

Computer Vision and Pattern Recognition are extremely important research fields with an enormous range of applications. They are also extremely difficult. This may seem paradoxical since humans can easily interpret images and detect spatial patterns. But this apparent ease is misleading because neuroscience shows that humans devote a large part of their brain, possibly up to fifty percent of the cortex, to processing images and interpreting them. The difficulties of these problems has been appreciated over the last thirty years as researchers have struggled to develop computer algorithms for performing vision and pattern recognition tasks. Although these problems are not yet completely solved it is becoming clear that the final theory will depend heavily on probabilistic techniques and the use of concepts from information theory.

The connections between information theory and computer vision are deep. Vision can be considered to be a decoding problem where the encoding of the information is performed by the physics of the world — by light rays striking objects and being reflected to cameras or eyes. Ideal observer theories were pioneered by scientists like Barlow [1] to compute the amount of information available in the visual stimuli and to see how efficient humans are at exploiting it. This leads to a research program where the goal is to develop probabilistic models which are capable of capturing the richness of visual stimuli and hence are efficient at encoding them.

This paper provides an introduction to these probabilistic models and discusses issues such as how to learn these models from natural stimuli and how to perform inference (i.e., decode and interpret the image). We will also describe how concepts, measures, and techniques from information theory can be applied to vision. For example, concepts like entropy and conditional entropy have been used for learning models and designing inference algorithms. This relates to Amari's work on information geometry [2,3] and Xu's concept of the Ying-Yang machine [4]. Similarly information theoretic measures and techniques, like Chernoff information and the theory of types, have been applied to analyze the performance of visual theories. Finally, we provide pointers to other uses of information theory ideas in the computer vision literature and draw attention to a recent book [5].

The structure of this paper is as follows. Section 2 discusses how information theory relates to pattern theory and Section 3 introduces minimum description length ideas. Section 4 describes Markov Random Field (MRF) models for images and Section 5 discusses inference and learning algorithms for MRFs. Section 6 described recent advanced image models which are better able to

Received April 19, 2010; accepted April 29, 2010

Alan YUILLE (✉)
Department of Statistics, University of California at Los Angeles, Los Angeles, CA 90095, USA
E-mail: yuille@stat.ucla.edu

deal with the complexities of natural images. Finally, Section 7 reviews how measures and techniques from information theory can be used to analyze the performance of vision models.

2 Information theory and pattern theory

Standard information theory [6,7] specifies how to encode data which obeys a probability distribution and encode it so that it can be transmitted and then decoded. For computer vision, however, the "encoding" is performed by light rays which are reflected off objects in the visual scene and transmitted to our eye or to a camera. Hence the encoding depends on the reflectance properties of objects, their spatial locations, and the positions of the light sources — all of this is out of our control. Nevertheless, most images have structures which suggest natural ways to encode them.

For example, consider the images shown in Fig. 1. Clearly there is an efficient way to represent the left image (a Kanizsa triangle [8]) in terms of one triangle in front of a second triangle and three black circles. Studies show that human observers perceive this interpretation despite the partial occlusions of the circles and the second triangle. Indeed human observers hallucinate that the surface of the triangle is brighter than the background and perceive sharp boundaries to the first triangle even at places where there is no direct visual cues. Note that there is an alternative simple interpretation for this image — three pacman figures aligned with three partial triangles — but this interpretation is not natural for human observers. The second image, the Dalmatian dog [9], initially seems merely to consist of a large number of randomly positioned dots. Closer inspection, however, shows that the image consists of a Dalmatian dog walking on a flat surface, which gives a more efficient interpretation. The third example is a more natural image, which be interpreted in terms of its constituent objects including text and faces.

These three examples suggest that we can achieve a tremendous amount of data compression by interpreting images in terms of the structure of the visual scene. They suggest a succession of increasingly more compact and semantically more meaningful interpretations. Studies of human vision suggest that these representations may be organized hierarchically in visual areas like V1, V2, V4 and IT [10]. Indeed, the principles of efficient coding and maximization of information transmission have been fruitful for obtaining quantitative theories to explain the behavior of neurons in the early stages of the visual system. These theories explain linear receptive field development, and various adaptation and normalization phenomena observed in these early areas [10–12].

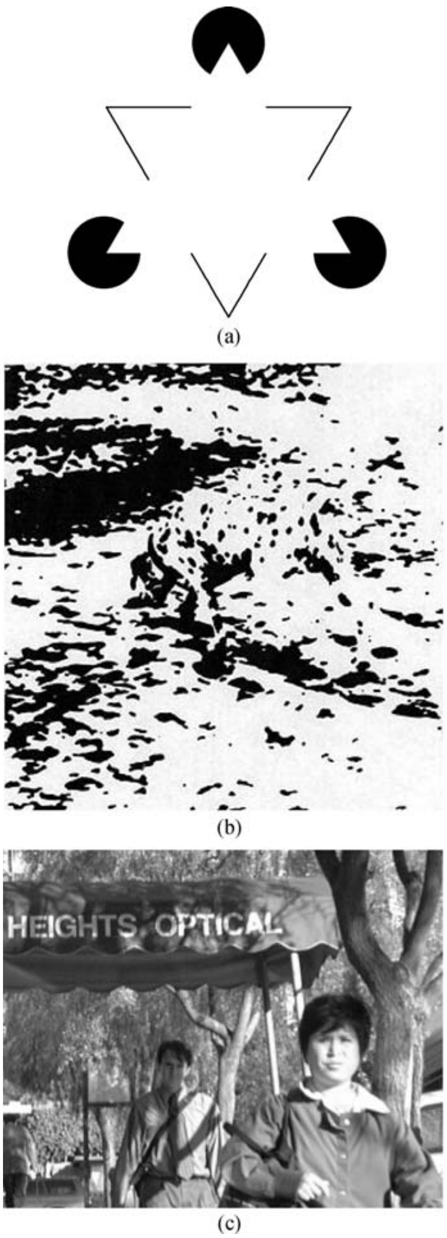


Fig. 1 Examples that illustrate how images are interpreted to make descriptions simpler and shorter. The Kanizsa triangle (a) can be compactly described as two triangles in front of three circles. Figure (b) is best interpreted as a dog although at first it appears to be a set of random blobs. The street scene (c) is best described in terms of objects like faces and text

But how do we represent and exploit these image structures? Researchers in pattern theory have advocated formulating vision as probabilistic inference on structured probability representations. This seems both a natural way in which to deal with the complexities and ambiguities of image patterns [13] and also fits into a more unified framework for cognition and artificial intelligence [14]. But vision is a particularly challenging problem to formulate in this manner. The complexity of vision seems to require distributions defined over very complicated structures and requires principles such as compositionality and the use of graphical models with variable topology [15,16]. In particular, Zhu and Mumford [16] have proposed a framework for modeling visual patterns using generative models which is illustrated by many real world examples.

3 Minimal length encoding

We discuss these models using Leclerc's perspective which formulates scene segmentation as an inference problem in terms of efficient encoding [17]. This approach is based on the Minimum Description Length (MDL) principle [18]. The computational goal is to choose the representation $\mathbf{W}$ of the regions best fits the image data $\mathbf{I}$, or equivalently, which best encodes the data. In Bayesian terms, we seek to perform Maximum a Posteriori (MAP) estimation by maximizing the *a posteriori* distribution $P(\mathbf{W}|\mathbf{I})$ of the representation conditioned on the data. By Bayes theorem, we can express this in terms of the likelihood function $P(\mathbf{I}|\mathbf{W})$ and the prior $P(\mathbf{W})$ as follows:

$$P(\mathbf{W}|\mathbf{I}) = \frac{P(\mathbf{I}|\mathbf{W})P(\mathbf{W})}{P(\mathbf{I})}.$$

The likelihood function $P(\mathbf{I}|\mathbf{W})$ specifies the probability of observing data $\mathbf{I}$ if the true representation is $\mathbf{W}$ and $P(\mathbf{W})$ is the prior probability of the representation (before the data). For example, in the weak-membrane model the likelihood function is a simple noise distribution and the prior encodes assumptions that the image is piecewise smooth and the boundaries are spatially smooth (see the next section for details).

In order to relate MAP estimation to efficient encoding, we take the logarithm of Bayes rule $\log P(\mathbf{W}|\mathbf{I}) = \log P(\mathbf{I}|\mathbf{W}) + \log P(\mathbf{W}) - \log P(\mathbf{I})$. $P(\mathbf{I})$ is constant (independent of $\mathbf{W}$), so MAP estimation corresponds to *minimizing* the encoding cost:

$$-\log P(\mathbf{I}|\mathbf{W}) - \log P(\mathbf{W}).$$

We now interpret this in terms of minimal encoding. By information theory [6,7] the number of bits required to encode a variable $\mathbf{W}$ which has probability distribution $P(\mathbf{W})$ is $-\log P(\mathbf{W})$. The term $-\log P(\mathbf{W})$ is the cost of encoding the interpretation $\mathbf{W}$. The term $-\log P(\mathbf{I}|\mathbf{W})$ is the cost of encoding the data $\mathbf{I}$ conditioned on interpretation $\mathbf{W}$. This cost will be 0 if the interpretation explains the data perfectly (i.e., $P(\mathbf{I}|\mathbf{W}) = 1$). But usually the interpretation will only partially explain the data and so $-\log P(\mathbf{I}|\mathbf{W})$ is called the residual (see the detailed example below).

Observe that the encoding depends on our choice of models $P(\mathbf{W}|\mathbf{I})$ and $P(\mathbf{W})$. Different models will lead to different encoding, as we will describe later.

4 Markov random field models for images

We now present Markov Random Field (MRF) models of images. These were the first type of probabilistic models used to describe images and they remain a good starting point into the literature. We will discuss the Potts model of images and the weak membrane model. The critical aspects of both models are: i) the *representation* which consists of the graph structure of the model and the state variables, ii) the *inference algorithm* used to estimate properties of the model such as the most probable state, and iii) the *learning algorithm* used to learn the parameters of the model. In earlier models learning was often not used and instead models were hand designed.

Firstly, we discuss *the Potts model* which is used for tasks such as image labeling and segmentation [19]. We will concentrate on the image labeling task where the goal is to assign a label to every image pixel $\mu \in \mathcal{D}$, where $\mathcal{D}$ is the image lattice. The input is an image $\mathbf{I}$, where $\mathbf{I} = \{I_\mu : \mu \in \mathcal{D}\}$ specifies the intensity values $I_\mu \in \{0, 255\}$ on the lattice, and the output $\mathbf{W}$ is $\mathbf{W} = \{w_\mu : \mu \in \mathcal{D}\}$ is a set of image labels $w_\mu \in \mathcal{L}$, see Fig. 2. The nature of the labels will depend on the problem. For edge detection, $|\mathcal{L}| = 2$ and the labels l_1, l_2 will correspond to 'edge' and 'non-edge'. For labeling the MSRC dataset [20] $|\mathcal{L}| = 23$ and the labels $l_1, \ldots, l_{23}$ include 'sky', 'grass', and so on. Similar models can be applied to other vision problems such as binocular stereo [21,22] (by setting the input to be the images to the left and right eyes $(\mathbf{I}^L, \mathbf{I}^R)$ and setting $\mathbf{w}$ to be the disparity labels).

We can *represent* the image labeling problem in terms of a probability distribution defined on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the set of nodes $\mathcal{V}$ is the set of image pixels $\mathcal{D}$ and the edges $\mathcal{E}$ are between neighboring pixels — see

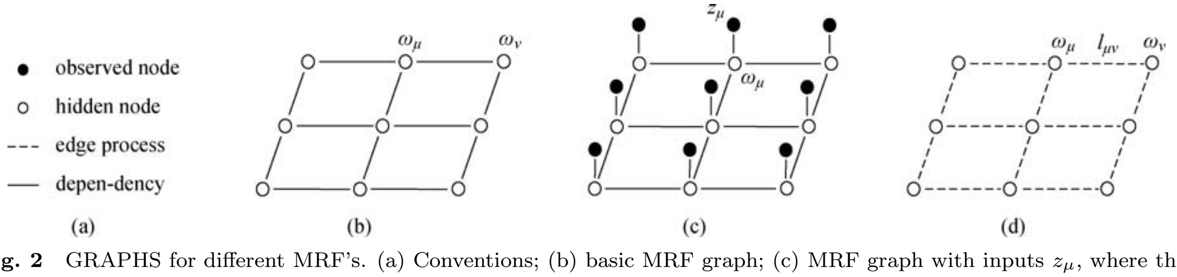


Fig. 2 GRAPHS for different MRF's. (a) Conventions; (b) basic MRF graph; (c) MRF graph with inputs z_μ, where the z's represent the filtered image; (d) a weak membrane model with line processes

Fig. 2. The $\mathbf{W} = \{w_\mu : \mu \in \mathcal{V}\}$ are random variables specified at each node of the graph. $P(\mathbf{W}|\mathbf{I})$ is a Gibbs distribution specified by an energy function $E(\mathbf{W}, \mathbf{I})$ which contains unary potentials $U(\mathbf{W}, \mathbf{I}) = \sum_{\mu \in \mathcal{V}} \boldsymbol{\lambda}^D \cdot \boldsymbol{\phi}(w_\mu, \mathbf{I})$ and pairwise potentials $V(\mathbf{W}, \mathbf{W}) = \sum_{\mu\nu \in \mathcal{E}} \boldsymbol{\lambda}^P \cdot \boldsymbol{\psi}_{\mu\nu}(w_\mu, w_\nu)$. The unary potentials $\boldsymbol{\phi}(w_\mu, \mathbf{I})$ depend only on the label/disparity at node/pixel μ and the input image $\mathbf{I}$. In practice, these models act on a filtered image which we represent by $\{z_\mu\}$ where the filters (e.g., Gabor filters) enhance important aspects of the image. The pairwise potentials impose prior assumptions about the local 'context' of the labels and disparities. These models typically assume that neighboring pixels will tend to have similar labels/disparities.

The full distribution $P(\mathbf{W}|\mathbf{I})$ defined over discrete-valued random variables $\mathbf{W} = \{w_\mu : \mu \in \mathcal{V}\}$ defined on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$:

$$P(\mathbf{W}|\mathbf{I}) = \frac{1}{Z(\mathbf{I})} \exp\left\{ -\sum_{\mu \in \mathcal{V}} \boldsymbol{\lambda}^D \cdot \boldsymbol{\phi}_\mu(w_\mu, \mathbf{I}) - \sum_{\mu\nu \in \mathcal{E}} \boldsymbol{\lambda}^P \cdot \boldsymbol{\psi}_{\mu\nu}(w_\mu, w_\nu) \right\}. \tag{1}$$

Secondly, we describe the *weak membrane model* which has applications to image segmentation and image denoising. It was proposed independently by Geman and Geman [19] and Blake and Zisserman [23]). This model has additional 'hidden variables' $\mathbf{L}$, which are used to explicitly label discontinuities. It is also a generative model which specifies a likelihood function and a prior probability (by contrast to conditional random fields which specify the posterior distribution only).

The input to the weak membrane model is the set of intensity values $\mathbf{I} = \{I_\mu : \mu \in \mathcal{D}\}$ and the output is $\mathbf{W} = \{w_\mu : \mu \in \mathcal{D}\}$ defined on a corresponding output lattice (formally we should specify two different lattices, say $\mathcal{D}_1$ and $\mathcal{D}_2$, but this makes the notation too cumbersome). We define a set of edges $\mathcal{E}$ which connect neighbouring pixels on the output lattice and define the set of line processes $\mathbf{L} = \{l_{\mu\nu} : (\mu, \nu) \in \mathcal{D}_e\}$ with $l_\mu \in \{0, 1\}$ over these edges, see Fig. 2. The weak membrane is a generative model so it is specified by two probability distributions: i) the likelihood function $P(\mathbf{I}|\mathbf{W})$, which specifies how the observed image $\mathbf{I}$ is a corrupted version of the image $\mathbf{W}$, and ii) the prior distribution $P(\mathbf{W}, \mathbf{L})$ which imposes a *weak membrane* by requiring that neighbouring pixels take similar values except at places where the line process is activated.

The simplest version of the weak membrane model is specified by the distributions:

$$P(\mathbf{I}|\mathbf{W}) = \prod_{\mu \in \mathcal{D}} \sqrt{\frac{\tau}{\pi}} \exp\{-\tau(I_\mu - w_\mu)^2\},$$
$$P(\mathbf{W}, \mathbf{L}) \propto \exp\{-E(\mathbf{W}, \mathbf{L})\},$$

$$\text{with } E(\mathbf{W}, \mathbf{L}) = A \sum_{(\mu,\nu) \in \mathcal{E}} (w_\mu - w_\nu)^2 (1 - l_{\mu\nu}) + B \sum_{(\mu,\nu) \in \mathcal{E}} l_{\mu\nu}. \tag{2}$$

In this model the intensity variables I_μ, w_μ are continuous-valued while the line processor variables $l_{\mu\nu} \in \{0, 1\}$, where $l_{\mu\nu} = 1$ means that there is an (image) edge at $\mu\nu \in \mathcal{E}$. The likelihood function $P(\mathbf{I}|\mathbf{W})$ assume independent zero-mean Gaussian noise (for other noise models, like shot noise, see Geiger and Yuille [24] and Black and Rangarajan [25]). The prior $P(\mathbf{W}, \mathbf{L})$ encourages neighboring pixels μ, ν to have similar intensity values $w_\mu \approx w_\nu$ except if there is an edge $l_{\mu\nu} = 1$. This prior imposes piecewise smoothness, or weak smoothness, which is justified by statistical studies of intensities and depth measurements (see Zhu and Mumford [26], Black and Roth [27]). More advanced variants of this model will introduce higher order coupling terms of form $l_{\mu\nu} l_{\rho\tau}$ into the energy $E(\mathbf{W}, \mathbf{L})$ to encourage edges to group into longer segments which may form closed boundaries. We can also re-express this model as $P(\mathbf{W}|\mathbf{I}) = \sum_{\mathbf{L}} P(\mathbf{W}, \mathbf{L}|\mathbf{I})$.

The *inference task* is to estimate the best interpretation of the input image $\mathbf{I}$. This corresponds to estimating the best "weakly smoothed" image or the image labels and the edges in the image. This is performed by specifying an inference algorithm to compute the MAP estimator:

$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} P(\mathbf{W}|\mathbf{I}). \tag{3}$$

The inference task is challenging for both models. We will describe some of these algorithms in the next section — choosing those that relate to information theoretic concepts.

The *learning task* is to learn the parameters of the distributions of the models from a set of training examples $\{(\mathbf{I}^i, \mathbf{W}^i) : i = 1, \ldots, N\}$. In early applications the parameters were set by hand — the weak membrane model is entirely set by hand. More recently, methods have been developed to learn the models. These will be described in Section 5 concentrating on methods that use information theoretic concepts.

5 Inference and learning for MRFs

A range of algorithms have been proposed but convergence guarantees are rare. Max-flow/min-cut [28] algorithms are guaranteed to converge to the optimal solution for certain classes of models if the state variables are binary-valued. If we allow the state variables $\mathbf{x}$ to take continuous values then steepest descent, and related methods, will also converge to the optimal estimate provided the energy function $\sum_{\mu\in\mathcal{V}} \boldsymbol{\lambda}^D \cdot \boldsymbol{\phi}_\mu(w_\mu, \mathbf{I}) + \sum_{\mu\nu\in\mathcal{E}} \boldsymbol{\lambda}^P \cdot \boldsymbol{\psi}_{\mu\nu}(w_\mu, w_\nu)$ is convex in the state variables $\mathbf{W}$. Markov Chain Monte Carlo (MCMC) methods are guaranteed to converge to good estimate of $\hat{\mathbf{W}}$, but convergence rates tend to be slow [19]. Other algorithms that empirically give good results for these types of models include variational mean field methods [24,29] and belief propagation [30]. This section will concentrate on variational and belief propagation algorithms, because they are formulated in terms of information theoretic concepts.

The *learning task* is to learn the model parameters $\boldsymbol{\lambda}$ from a set of supervised training examples. The learning is straightforward if we only consider only the unary potentials, because we can learn the data parameters $\boldsymbol{\lambda}^D$ by methods such as AdaBoost [31] or simply by learning conditional probability distributions [32]. Discriminative learning methods [33] have been used to learn the full distribution, which requires an efficient inference algorithm, so that we can compare the performance of the algorithm with its current set of parameters to the groundtruth, and then modify the parameters if necessary. This can be formulated in terms of maximum likelihood learning or conditional random fields. In this paper we will restrict ourselves to describing the minimax entropy method for learning binary, and higher order, potential terms [34].

5.1 Inference for Potts model

For the Potts model we will describe techniques based on minimizing free energies [35]. We first describe the mean field free energy approach [24, 29, 36, 37] and then describe the Bethe free energy which yields belief propagation [30]. We refer to Wainwright et el. [38] for the related convex free energies. In this section, for simplicity, we will drop the parameters $\boldsymbol{\lambda}$ of the potentials and express them by $\boldsymbol{\phi}(\cdot)$ and $\boldsymbol{\psi}(\cdot)$ only.

The basic idea of *Mean Field Theory* is to approximate a distribution $P(\mathbf{W}|\mathbf{I})$ by a simpler distribution $B(\mathbf{W}|\mathbf{I})$ which is chosen so that it is easy to estimate (approximately) the MAP estimate of $P(\cdot)$, and any other estimator, from the approximate distribution $B(\cdot)$. This requires specifying a class of approximating distributions $\{B(\cdot)\}$, a measure of similarity between distributions $B(\cdot)$ and $P(\cdot)$, and an algorithm for finding the $B^*(\cdot)$ that minimizes the similarity measure. In this paper we restrict the class of approximating distributions to be factorizable $B(\mathbf{W}) = \prod_{\mu\in\mathcal{V}} b_\mu(w_\mu)$. The measure of similarity is the *mean field free energy* which can be expressed as $\mathcal{F}_{\mathrm{MFT}}(B)$:

$$\begin{aligned}\mathcal{F}_{\mathrm{MFT}}(B) = &\sum_{\mu\nu\in\mathcal{E}} \sum_{w_\mu, w_\nu} b_\mu(w_\mu) b_\nu(w_\nu) \psi_{\mu\nu}(w_\mu, w_\nu)\\ &+ \sum_{\mu\in\mathcal{V}} \sum_{w_\mu} b_\mu(w_\mu) \phi_\mu(w_\mu, \mathbf{I})\\ &+ \sum_{\mu\in\mathcal{V}} \sum_{w_\mu} b_\mu(w_\mu) \log b_\mu(w_\mu). \end{aligned} \tag{4}$$

The first two terms are the expectation of the energy $E(\mathbf{W}, \mathbf{I})$ with respect to the distribution $B(\mathbf{W})$ and the third term is the negative entropy of $B(\mathbf{W})$.

The most intuitive derivation of the mean field free energy is obtained by minimizing the Kullback-Leibler divergence between the target distribution $P(\mathbf{W}|\mathbf{I})$ and a factorizable trial distribution $B(\mathbf{W})$ [37]. Substituting $P(\mathbf{W}) = \dfrac{1}{Z} \exp\{-E(\mathbf{W}, \mathbf{I})\}$ and $B(\mathbf{W}) = \prod_{\mu\in\mathcal{V}} b_\mu(w_\mu)$ into the Kullback-Leibler divergence $KL(B, P)$ gives:

$$\begin{aligned} KL(B, P) &= \sum_{\mathbf{W}} B(\mathbf{W}) E(\mathbf{W})\\ &\quad + \sum_{\mathbf{W}} B(\mathbf{W}) \log B(\mathbf{W}) + \log Z\\ &= \mathcal{F}_{\mathrm{MFT}}(B) + \log Z. \end{aligned} \tag{5}$$

Hence minimizing $\mathcal{F}_{\mathrm{MFT}}(B)$ with respect to B gives: i) the best factorized approximation to $P(\mathbf{W})$, and ii) a lower bound to the partition function $\log Z \geqslant \min_B \mathcal{F}_{\mathrm{MFT}}(B)$ which can be useful to assess model evidence [39].

Minimizing the mean field free energy can be performed by steepest descent techniques. The free energy can have local minima so there is no guarantee of optimality. But the approach works well in practice on a range of vision problems.

The *Bethe free energy* [40] differs from the MFT free energy by including pairwise pseudo-marginals $b_{\mu\nu}(w_\mu, w_\nu)$:

$$\begin{aligned}\mathcal{F}[b; \lambda] = &\sum_{\mu\nu} \sum_{w_\mu, w_\nu} b_{\mu\nu}(w_\mu, w_\nu) \psi_{\mu\nu}(w_\mu, w_\nu)\\ &+ \sum_{\mu} \sum_{w_\mu} b_\mu(w_\mu) \phi_\mu(w_\mu)\\ &+ \sum_{\mu\nu} \sum_{w_\mu, w_\nu} b_{\mu\nu}(w_\mu, w_\nu) \log b_{\mu\nu}(w_\mu, w_\nu)\\ &- \sum_{\mu} (n_\mu - 1) \sum_{w_\mu} b_\mu(w_\mu) \log b_\mu(w_\mu). \end{aligned} \tag{6}$$

where n_μ is the number of neighbours of node μ. We must also impose consistency and normalization constraints which we impose by lagrange multipliers $\{\lambda_{\mu\nu}(w_\nu)\}$ and $\{\gamma_\mu\}$:

$$\begin{aligned} &\sum_{\mu,\nu} \sum_{w_\nu} \lambda_{\mu\nu}(w_\nu) \left\{ \sum_{w_\mu} b_{\mu\nu}(w_\mu, w_\nu) - b_\nu(w_\nu) \right\}\\ &+ \sum_{\mu,\nu} \sum_{w_\mu} \lambda_{\nu\mu}(w_\mu) \left\{ \sum_{w_\nu} b_{\mu\nu}(w_\mu, w_\nu) - b_\mu(w_\mu) \right\} \end{aligned}$$

$$+\sum_{\mu}\gamma_{\mu}\left\{\sum_{w_{\mu}}b_{\mu}(w_{\mu})-1\right\}. \tag{7}$$

The *belief propagation* algorithm (BP) is defined in terms of messages $m_{\mu\nu}(w_{\nu})$ from μ to ν, and is specified by the sum-product update rule:

$$m_{\mu\nu}^{t+1}(w_{\nu})=\sum_{w_{\mu}}\exp\{-\psi_{\mu\nu}(w_{\mu},w_{\nu})-\phi_{\mu}(w_{\mu})\}\cdot\prod_{\rho\neq\nu}m_{\rho\mu}^{t}(w_{\mu}). \tag{8}$$

The unary and binary pseudomarginals are related to the messages by

$$b_{\mu}^{t}(w_{\mu})\propto\exp\{-\phi_{\mu}(w_{\mu})\}\prod_{\rho}m_{\rho\mu}^{t}(w_{\mu}), \tag{9}$$

$$b_{\rho\nu}^{t}(w_{\rho},w_{\nu})\propto\exp\{-\psi_{\rho\nu}(w_{\rho},w_{\nu})-\phi_{\rho}(w_{\rho})-\phi_{\nu}(w_{\nu})\}\times\prod_{\tau\neq j}m_{\tau\rho}^{t}(w_{\rho})\prod_{\zeta\neq\rho}m_{\zeta\nu}^{t}(w_{\nu}). \tag{10}$$

It can be shown [30] that fixed points of BP are extrema of the Bethe free energy. Hence BP can be used to find pseudomarginals $\{b\}$ that will give good approximations to the target distribution $P(\mathbf{W}|\mathbf{I})$. But the update rule for BP is not guaranteed to converge to a fixed point for general graphs and can sometimes oscillate wildly. It can be partially stabilized by adding a damping term to Equation (8) — e.g., by multiplying the right hand side by $(1-\epsilon)$ and adding a term $\epsilon m_{\mu\nu}^{t}(w_{\nu})$.

5.2 Inference for weak membrane model

We can obtain free energies for the weak membrane model also. We use pseudo-marginals $B(\mathbf{L})$ for the line processes $\mathbf{L}$ only. This leads to a free energy $\mathcal{F}_{\mathrm{MFT}}(B,\mathbf{W})$ specified by

$$\mathcal{F}_{\mathrm{MFT}}(B,\mathbf{W})=\tau\sum_{\mu\in\mathcal{V}}(w_{\mu}-I_{\mu})^{2}+A\sum_{\mu\nu\in\mathcal{E}}(1-b_{\mu\nu})(w_{\mu}-w_{\nu})^{2}+B\sum_{\mu\nu\in\mathcal{E}}b_{\mu\nu}+\sum_{\mu\nu\in\mathcal{E}}\{b_{\mu\nu}\log b_{\mu\nu}+(1-b_{\mu\nu})\log(1-b_{\mu\nu})\}, \tag{11}$$

where $b_{\mu\nu}=b_{\mu\nu}(l_{\mu\nu}=1)$ (the derivation uses the fact that $\sum_{l_{\mu\nu}=0}^{1}b_{\mu\nu}(l_{\mu\nu})l_{\mu\nu}=b_{\mu\nu}$). As described below, this free energy is exact and involves no approximations.

For the weak membrane model the free energy follows from Neal and Hinton's variational formulation of the expectation maximization EM algorithm [41]. The goal of EM is to estimate $\mathbf{W}$ from $P(\mathbf{W}|\mathbf{I})=\sum_{\mathbf{L}}P(\mathbf{W},\mathbf{L}|\mathbf{I})$ after treating the $\mathbf{L}$ as 'nuisance variables' which should be summed out [39]. This can be expressed [41] in terms of minimizing the free energy function:

$$\mathcal{F}_{\mathrm{EM}}(B,\mathbf{W})=-\sum_{\mathbf{L}}B(\mathbf{L})\log P(\mathbf{W},\mathbf{L}|\mathbf{I})+\sum_{\mathbf{L}}B(\mathbf{L})\log B(\mathbf{L}). \tag{12}$$

The equivalence of minimizing $\mathcal{F}_{\mathrm{EM}}[B,\mathbf{W}]$ and estimating $\mathbf{W}^{*}=\arg\max_{\mathbf{W}}P(\mathbf{W}|\mathbf{I})$ can be verified by re-expressing $\mathcal{F}_{\mathrm{EM}}[B,\mathbf{W}]$ as $-\log P(\mathbf{W}|\mathbf{I})+\sum_{\mathbf{L}}B(\mathbf{L})\log\dfrac{B(\mathbf{L})}{P(\mathbf{L}|\mathbf{W},\mathbf{I})}$, from which it follows that the global minimum occurs at $\mathbf{W}^{*}=\arg\min_{\mathbf{W}}\{-\log P(\mathbf{W}|\mathbf{I})\}$ and $B(\mathbf{L})=P(\mathbf{L}|\mathbf{W}^{*},\mathbf{I})$ (because the second term is the Kullback-Leibler divergence which is minimized by setting $B(\mathbf{L})=P(\mathbf{L}|\mathbf{W},\mathbf{I})$).

The EM algorithm minimizes $\mathcal{F}_{\mathrm{EM}}[B,\mathbf{W}]$ with respect to B and $\mathbf{W}$ alternatively, which gives the E-step and the M-step respectively. For the basic weak membrane model both steps of the algorithm can be performed simply. The E-step requires minimizing a quadratic function, which can be performed by linear algebra, while the M-step can be computed analytically:

$$\text{Minimize}\left\{\sum_{\mu}\tau(\mathrm{w}_{\mu}-\mathrm{I}_{\mu})^{2}+\mathrm{A}\sum_{\mu\nu\in\mathcal{E}}\mathrm{b}_{\mu\nu}(\mathrm{w}_{\mu}-\mathrm{w}_{\nu})^{2}\right\}\text{ wrt }\mathbf{W}, \tag{13}$$

$$B(\mathbf{L})=\prod_{\mu\nu\in\mathcal{E}}b_{\mu\nu}(l_{\mu\nu}),\quad b_{\mu\nu}=\frac{1}{1+\exp\{-A(w_{\mu}-w_{\nu})^{2}+B\}}. \tag{14}$$

The EM algorithm is only guaranteed to converge to a local minimum of the free energy and so good choices of initial conditions are needed. A natural initialization for the weak membrane model is to set $\mathbf{W}=\mathbf{I}$, perform the E-step, then the M-step, and so on. Observe that the M-step corresponds to performing a weighted smoothing of the data $\mathbf{I}$ where the smoothing weights are determined by the current probabilities $B(\mathbf{L})$ for the edges. The E-step estimates the probabilities $B(\mathbf{L})$ for the edges given the current estimates for the $\mathbf{W}$.

Notice that the EM free energy does not put any constraints of the form of the distribution B and yet the algorithm results in a factorized distribution, see Equation (14). This results naturally because the variables that are being summed out — the $\mathbf{L}$ variables — are conditionally independent (i.e., there are no terms in the energy $E(\mathbf{W},\mathbf{I})$ which couple $y_{\mu\nu}$ with its neighbors). In addition we can compute $P(\mathbf{W}|\mathbf{I}) = \sum_{\mathbf{L}} P(\mathbf{W},\mathbf{L}|\mathbf{I})$ analytically to obtain $\frac{1}{Z}\exp\{-\tau\sum_{\mu\in\mathcal{D}}(w_\mu - z_\mu)^2 - \sum_{\mu\nu\in\mathcal{E}} g(w_\mu - w_\nu)\}$, where $g(w_\mu - w_\nu) = -\log\{\exp\{-A(w_\mu - w_\nu)^2\} + \exp\{B\}\}$. The function $g(w_\mu - w_\nu)$ penalizes $w_\mu - w_\nu$ quadratically for small $w_\mu - w_\nu$ but tends to a finite value asymptotically for large $|w_\mu - w_\nu|$.

Suppose, however, that we consider a modified weak membrane model which includes interactions between the line processes — terms in the energy like $C\sum_{(\mu\nu)\times(\rho\zeta)\in\mathcal{E}_y} l_{\mu\nu}l_{\rho\zeta}$ which encourage lines to be continuous. It is now impossible either to: a) solve for $B(\mathbf{L})$ in closed form for the E-step of EM, or b) to compute $P(\mathbf{W}|\mathbf{L})$ analytically. Instead we use the mean field approximation by requiring that B is factorizable — $B(\mathbf{L}) = \prod_{\mu\nu\in\mathcal{E}} b_{\mu\nu}(l_{\mu\nu})$. This gives a free energy:

$$\begin{aligned}&\mathcal{F}_{\mathrm{MFT}}(\mathbf{b},\mathbf{W})\\ &= \tau\sum_{\mu\in\mathcal{V}}(w_\mu - I_\mu)^2 + A\sum_{\mu\nu\in\mathcal{E}}(1-b_{\mu\nu})(w_\mu - w_\nu)^2\\ &\quad + B\sum_{\mu\nu\in\mathcal{E}} b_{\mu\nu} + C\sum_{(\mu\nu)\times(\rho\zeta)\in\mathcal{E}} b_{\mu\nu}b_{\rho\zeta}\\ &\quad + \sum_{\mu\nu\in\mathcal{E}}\{b_{\mu\nu}\log b_{\mu\nu} + (1-b_{\mu\nu})\log(1-b_{\mu\nu})\}. \quad (15)\end{aligned}$$

5.3 Learning: Minimax entropy

Minimax entropy learning [34] is a technique for learning probability distributions from image statistics. When applied to natural images it learns distributions similar to the hand designed distributions in the weak membrane model [26].

Suppose we observe statistics $\boldsymbol{\phi}(\mathbf{I}) = \boldsymbol{\psi}$ from an image. These statistics can be the histogram responses of filters applied to the image. The *maximum entropy principle* proposes to select the distribution $P(\mathbf{I})$ which has maximum entropy but with fixed expected value of the statistics. This leads to an optimization problem — select $P(\mathbf{I})$ to maximize

$$\begin{aligned}&-\sum_{\mathbf{I}} P(\mathbf{I})\log P(\mathbf{I}) + \nu\left\{\sum_{\mathbf{I}} P(\mathbf{I}) - 1\right\}\\ &\quad + \boldsymbol{\lambda}\cdot\left\{\sum_{\mathbf{I}} P(\mathbf{I})\boldsymbol{\phi}(\mathbf{I}) - \boldsymbol{\psi}\right\}, \quad (16)\end{aligned}$$

where $\boldsymbol{\psi}$ is the observed value of the statistic $\boldsymbol{\phi}(\mathbf{I})$ when evaluated on the data, and $\nu, \boldsymbol{\lambda}$ are Lagrange multipliers.

This maximization leads to a distribution:

$$P(\mathbf{I}|\boldsymbol{\lambda}) = \frac{1}{Z[\boldsymbol{\lambda}]}\mathrm{e}^{\boldsymbol{\lambda}\cdot\boldsymbol{\phi}(\mathbf{I})}, \quad (17)$$

where the parameters $\boldsymbol{\lambda}$ are chosen to ensure that $\sum_{\mathbf{I}} P(\mathbf{I}|\boldsymbol{\lambda})\boldsymbol{\phi}(\mathbf{I}) = \boldsymbol{\psi}$.

Solving this equation for $\boldsymbol{\lambda}$ is often difficult. It reduces to minimizing the convex function $\log Z[\boldsymbol{\lambda}] - \boldsymbol{\lambda}\cdot\boldsymbol{\psi}$. Algorithms such as steepest descent and generalized iterative scaling (GIS) can do this, but they require estimating the expectation $\sum_{\mathbf{I}} P(\mathbf{I}|\boldsymbol{\lambda})\boldsymbol{\phi}(\mathbf{I})$ for different values of $\boldsymbol{\lambda}$ which can be time consuming. Although stochastic MCMC methods are adequate [34]. It should be noted that this is equivalent to doing maximum likelihood estimation of the parameters $\boldsymbol{\lambda}$ assuming that we have observations $\{\mathbf{I}_i : i = 1,\ldots,N\}$, which we assume are i.i.d. generated from distribution $P(\mathbf{I}|\boldsymbol{\lambda})$ and where $\boldsymbol{\psi} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{\phi}(\mathbf{I}_i)$.

The approach can be extended to selecting from a dictionary of features $\boldsymbol{\phi}_a(\cdot) : a \in \mathcal{A}$. We can combine features to obtain a distribution:

$$P(\mathbf{I}) = \frac{1}{Z[\boldsymbol{\lambda}_1,\ldots,\boldsymbol{\lambda}_M]}\exp\left\{-\sum_{a=1}^{M}\boldsymbol{\lambda}_a\cdot\boldsymbol{\phi}_a(\mathbf{I})\right\}. \quad (18)$$

The choice of which features to use can be determined by standard model selection techniques, see Ref. [39]. Zhu et al. [34] proposed a related criteria to select features based on minimum entropy, hence “mini” in minimax, and described a greedy procedure for selection.

In vision, the feature statistics are often the histogram of local filters, such as $\partial\mathbf{I}/\partial x, \partial\mathbf{I}/\partial y$. Let us refer to these filters as $f(\mathbf{I}_a)$ where a is the position in the image, and quantize the filters so that their response takes values $z = 1,\ldots,M$. We write the histogram response as

$$\phi(z;\mathbf{I}) = \frac{1}{N}\sum_{a=1}^{N}\delta_{f(\mathbf{I}_a),z}, \quad (19)$$

where δ is the Kronecker delta. We can then write the distribution as

$$\begin{aligned}P(\mathbf{I}|\boldsymbol{\lambda}) &= \frac{1}{Z}\mathrm{e}^{\sum_z \boldsymbol{\lambda}(z)\cdot\boldsymbol{\phi}(z;\mathbf{I})}\\ &= \frac{1}{Z}\mathrm{e}^{\sum_z \boldsymbol{\lambda}(z)(1/N)\sum_{a=1}^{N}\delta_{f(\mathbf{I}_a),z}}\\ &= \frac{1}{Z}\mathrm{e}^{(1/N)\sum_{a=1}^{N}\boldsymbol{\lambda}(f(\mathbf{I}_a))}. \quad (20)\end{aligned}$$

This is a Markov random field where the local dependencies between different pixel sites a is determined by the filters $f(\cdot)$.

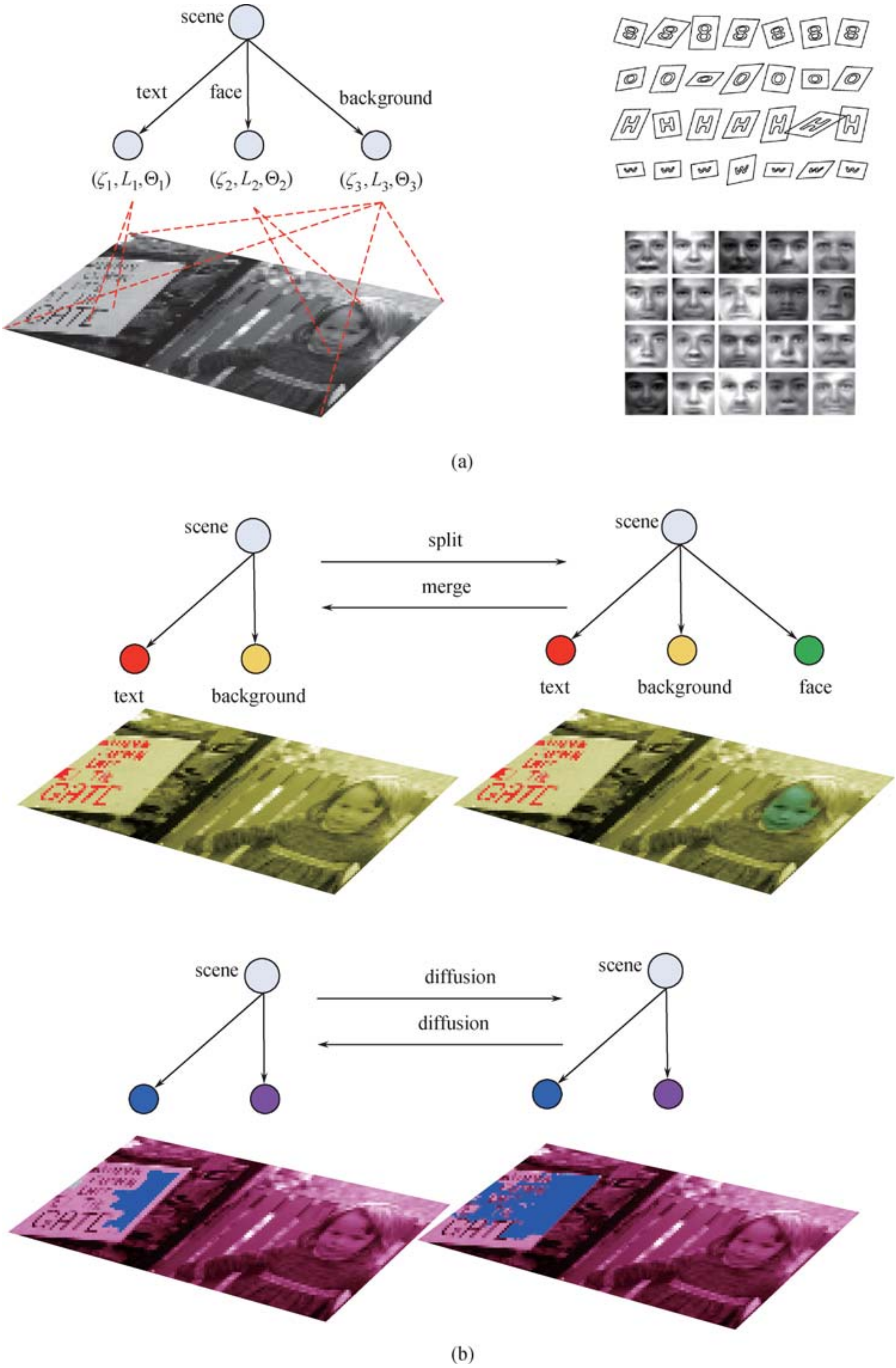
6 More advanced image models

We now describe more advanced models of images that have richer representations of images and, in particular, represent longer range structures. These lead to better encoding and interpretation of images. In particular, we will discuss imaging parsing models [16,42,43] and hierarchical models [44]. By necessity, these models are more complex than the MRF models described in the previous sections and hence they require more computation for inference and learning.

6.1 Image parsing

The goal of image parsing is to probabilistic models for image formation which are capable of generating all the complex visual patterns that occur in natural images. This is an ambitious and exciting research program which, in principle, is capable of solving all vision problems. The full research program involves stochastic grammars for images and is described in detail in Ref. [16]. In this paper, we will restrict ourselves to simpler models used in Refs. [42,43,45].

The basic idea of image parsing is illustrated in Fig. 3. We assume that regions in the image are generated by models of objects (e.g., faces, and text) and texture.



(a)

(b)

Fig. 3 (a) Ideas of image parsing; (b) “moves” for image parsing

The task of image parsing is to find the most probable way the image was generated. This requires a sophisticated inference algorithm which is called Data Driven Markov Chain Monte Carlo (DDMCMC) [42,43] which searches through alternative interpretations of images by a series of 'moves', see Fig. 3(b).

More formally, we assume that the image domain $\mathcal{D}$ can be decomposed into disjoint regions $\mathcal{D} = \bigcup_{a=1}^{M} \mathcal{D}_a$, with $\mathcal{D}_a \bigcap \mathcal{D}_b = \emptyset \ \forall \ a \neq b$ and where the number of regions M is a random variable.

We assume a class of probability models $P(\mathbf{I}_{\mathcal{D}_a}|\zeta_a, \theta_a)$ for representing the image intensity $\mathbf{I}_{\mathcal{D}_a}$ within each subregion $\mathcal{D}_a$, where ζ_a labels the model type and θ_a labels its parameters — e.g., ζ could label the region as 'texture' and θ would specify the parameters of the texture model. Other models include 'face' or 'digits'.

Hence the image is represented by $\mathbf{W} = (M, \{(\mathcal{D}_a, \zeta_a, \theta_a) : a = 1, \ldots, M\})$ and the likelihood model for the full image is of form:

$$P(\mathbf{I}|M, \{\mathcal{D}_a, \zeta_a, \theta_a\}) = \prod_{a=1}^{M} P(\mathbf{I}_a|\zeta_a, \theta_a). \qquad (21)$$

There is a prior probability on $\mathbf{W}$ specified by $P(M)P(\{\mathcal{D}_a\}|M)\prod_a P(\zeta_a)P(\theta_a)$. This specifies priors on the shapes of regions and on their image properties.

The goal of inference is to estimate

$$\begin{aligned}\hat{\mathbf{W}} = \arg\max P(\mathbf{I}|M, \{\mathcal{D}_a, \zeta_a, \theta_a\})P(M) \\ \cdot P(\{\mathcal{D}_a\}|M)\prod_a P(\zeta_a)P(\theta_a). \qquad (22)\end{aligned}$$

Inference. The inference algorithm has the difficult task of determining the number of image regions M, their shapes and positions $\{\partial D_a\}$, the model types $\{\zeta_a\}$ (e.g., face or texture) that generate them, and the parameters $\{\theta_a\}$ of the models.

The inference algorithm is illustrated in Fig. 3(b). It initializes a representation $\mathbf{W}$ of the image in terms of a number of regions, model types, and model parameters. Then it allows a set of 'moves' which alters the representation. These consist of: i) smoothly moving the boundary between two regions D_a and D_b, ii) splitting or merging a region: $(D_a) \leftrightarrow (D_b, D_c)$ or $(D_b, D_c) \leftrightarrow (D_a)$, iii) creating or deleting an object region, iv) altering the type ζ_a of a region D_a, v) changing the parameters θ_a of a region R_a.

We formulate a Markov Chain Monte Carlo (MCMC) algorithm [46] which uses these moves. Observe that the moves can be decomposed into two types: i) jumps which cause discrete changes to $\mathbf{W}$, and ii) diffusions which make continuous changes to $\mathbf{W}$. Firstly, *jump moves* include the birth/death of region hypotheses, the splitting and merging of regions, and changing the model type of a region (e.g., changing from a texture model to a text model), creating a face or a letter region. Secondly, *diffusion processes* which involve altering the boundary of a region and changing the parameters of the model which describes a region.

More formally, Data Driven Markov Chain Monte Carlo (DDMCMC) is a version of the Metropolis-Hastings algorithm. It uses data-driven proposal probabilities $q(\mathbf{W} \mapsto \mathbf{W}'|\mathbf{I})$ to make proposals which can be verified or rejected by the probability models $P(\mathbf{I}|\mathbf{W})P(\mathbf{W})$. Moves are selected by sampling from $q(\mathbf{W} \mapsto \mathbf{W}'|\mathbf{I})$ and they are accepted with probability $\alpha(\mathbf{W} \mapsto \mathbf{W}')$:

$$\alpha(\mathbf{W} \to \mathbf{W}') = \min\left(1, \frac{p(\mathbf{W}'|\mathbf{I})}{p(\mathbf{W}|\mathbf{I})} \cdot \frac{q(\mathbf{W}' \to \mathbf{W}|\mathbf{I})}{q(\mathbf{W} \to \mathbf{W}'|\mathbf{I})}\right). \qquad (23)$$

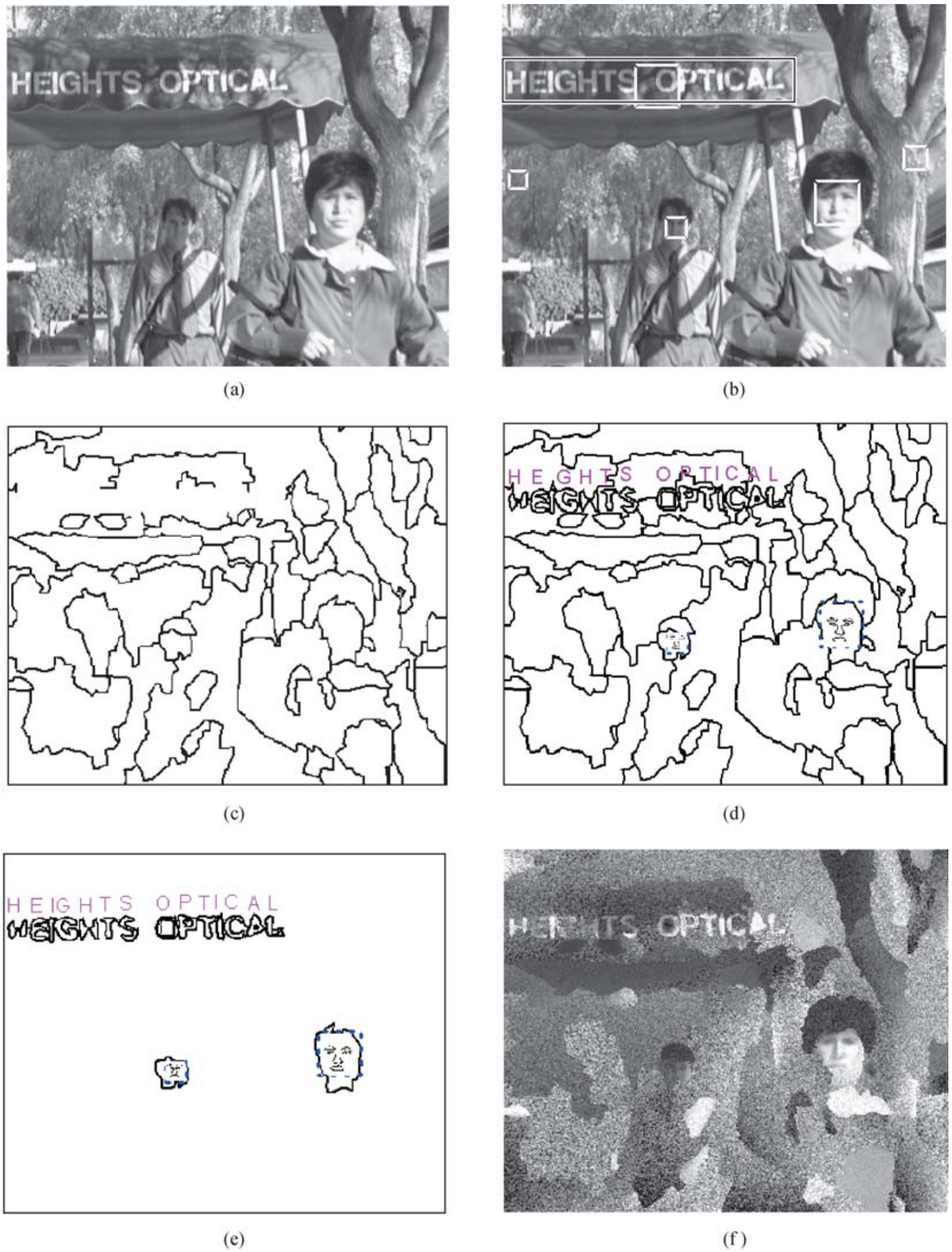
DDMCMC satisfies the necessary convergence conditions for an MCMC algorithm and so is guaranteed to converge to samples from the posterior distribution $P(\mathbf{W}|\mathbf{I}) \propto P(\mathbf{I}|\mathbf{W})P(\mathbf{W})$ [42,43]. In practice, the speed of convergence is determined by the effectiveness of the proposal probabilities $q(\mathbf{W} \leftarrow \mathbf{W}')$. Tu and Zhu [42] describe how the proposals are defined when the region models are 'generic' (i.e., do not include objects). When the types of models are extended to include objects, such as faces and textures, then additional proposals are used. For example, AdaBoost [47] was used to create proposals for faces [31]and text [48] where, following Hastie et al. [49], we interpret the output of AdaBoost as a conditional probability. Shape context features [50] can be used to make proposals for specific letters and digits within text regions. For more details, see Refs. [42, 43]. An example of the algorithm showing different stages is given by Fig. 4.

Learning the distributions $P(\mathbf{I}_{\mathcal{D}_a}|\tau_a, \gamma_a)$ is easier once images have been hand-labelled. For example, we used generative models of faces and text learnt from active appearance models [51] and shape models [52].

Results and later work

DDMCMC was successfully applied to image segmentation using 'generic image models' by Tu and Zhu [42]. It achieved the best performance on the Berkeley segmentation database [53] when evaluated in 2002. The work on image parsing with face and object models was not evaluated on a benchmarked dataset (none were available at the time) but individual components, e.g., the text detection [48] and shape matching (for letter and digit reading) [52] performed at the state of the art when evaluated on large datasets.

More recent work following this research program is described in a review paper by Zhu and Mumford [16]. It includes the primal sketch model [54] which describes the intensity changes at the boundaries between objects and regions and the use of generative grammars [55].



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 4 Results of the algorithm at different stages. (a) Input image; (b) proposals for text and faces illustrated by boxes (note the incorrect proposals in the bark of the tree and in the vegetation); (c) estimated boundaries of regions; (d) estimated boundaries and text; (e) detection of faces and text (observe that the incorrect proposals were rejected by the generative models); (f) a synthesized image which is sampled from the estimated model $\hat{\mathbf{W}}$

6.2 Hierarchical models: Image labeling. Segmentation-recognition templates

Hierarchical models give an alternative way to represent images, see Fig. 5 and Ref. [44]. They are based on the *compositionality hypothesis* that objects and images have structure at different scales and can be represented as recursive composition of parts, with the representational complexity roughly the same at all levels. This enables us to represent complex structures hierarchically using hidden variables and avoids the need for dense and long range sideways connections that flat models would require. These models use the *summarization principle* which requires that the complexity of the representation is the same at all levels of the hierarchy. This implies that hat upper levels of the hierarchy specify a coarse, or ‘executive summary’, representation of the object while the lower level nodes give more fine scale detail. This

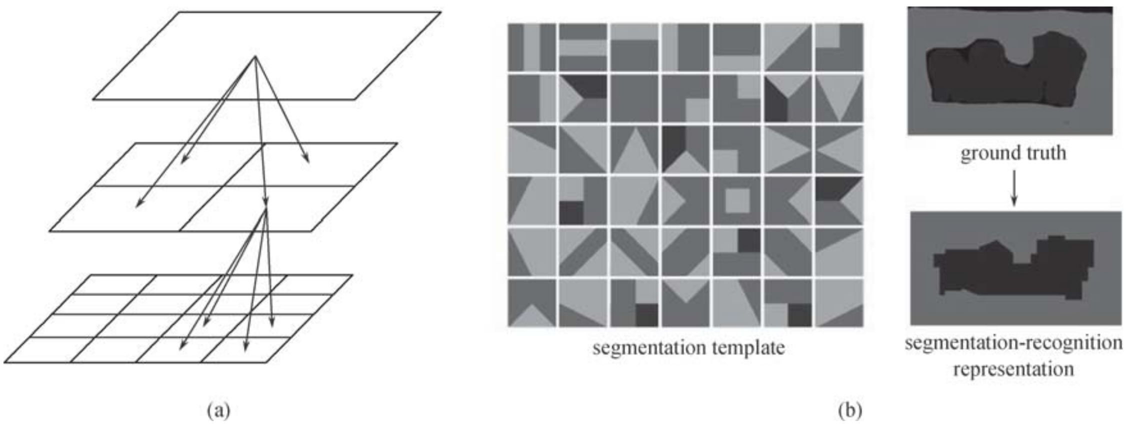


Fig. 5 (a) Structure of a hierarchical image model [44]. The grey circles are the nodes of the hierarchy. The graph structure is a fixed quadtree so that nodes have four parents; (b) 30 segmentation templates

principle enables efficient representation, inference, and learning. It has also been applied to modeling objects [56–58].

We define a hierarchical graph structure $(\mathcal{V}, \mathcal{E})$ with leaf nodes $\mathcal{V}^{\text{leaf}}$. The graph has a quadtree structure so that each node $\mu \in \mathcal{V}/\mathcal{V}^{\text{leaf}}$ has four child nodes $\nu \in ch(\mu)$, which defines the edges $\mathcal{E}$. The graph is organized in layers so that the nodes at each layer covers the image, see Fig. 5. We define state variables w_μ at each node, where $w_\mu = (s_\mu, \mathbf{c}_\mu)$ describes a subregion of the image. The variable s_μ indexes the partition of the image region into different sub-regions ($|\mathcal{S}| = 40$ and there are either one, two or three sub-regions), see Fig. 5. The variable $\mathbf{c}_\mu$ specifies labels for all the sub-regions where each label $c \in \mathcal{C}$, where $\mathcal{C}$ is a set of pre-defined labels ($|\mathcal{C}| = 23$ for the Cambridge Microsoft Labeled Dataset — [20]). We use the notation $\mathbf{w}_{ch(\mu)}$ to denote the states of all child nodes of μ.

We define a probability distribution $P(\mathbf{W}|\mathbf{I}) = \dfrac{1}{Z}\exp\{-\boldsymbol{\lambda} \cdot \boldsymbol{\Phi}(\mathbf{W}, \mathbf{I})\}$ in terms of potential functions $\boldsymbol{\Phi}(\mathbf{W}, \mathbf{I})$ are their parameters $\boldsymbol{\lambda}$ (which will be learnt). The potentials are of two types: i) data potentials $\boldsymbol{\phi}(w_\mu, \mathbf{I})$ which are defined for all nodes and which are functions of the image $\mathbf{I}$, and ii) prior potentials $\boldsymbol{\psi}(w_\mu, \mathbf{w}_{ch(\mu)})$ which impose statistical relations between the states of parents and child nodes — i.e., between the partitions and labels at neighboring levels. We use six potentials described in detail in Ref. [44]: i) data terms $\boldsymbol{\phi}^1_\mu(w_\mu, \mathbf{I})$ which represents image features of regions (features are color, Gabors, difference of Gaussians, etc.), ii) data terms $\boldsymbol{\phi}^2_\mu(w_\mu, \mathbf{I})$ which favors segmentation templates whose pixels within each partition have similar appearances, iii) prior terms $\boldsymbol{\psi}^3_\mu(w_\mu, \mathbf{w}_{ch(\mu)})$ which encourage consistency between the segmentation templates and labeling of parent and child nodes, iv) prior terms $\boldsymbol{\psi}^4_\mu(w_\mu, \mathbf{w}_{ch(\mu)})$ which captures the co-occurrence of different labels (e.g., a cow is unlikely to be next to an airplane), v) prior terms $\boldsymbol{\psi}^5_\mu(w_\mu)$ which puts probabilities on the segmentation templates s_μ, and iv) prior terms $\boldsymbol{\phi}^6_\mu(w_\mu)$ which capture the co-occurrence of the labels and the segmentation templates.

The full energy is defined by

$$\begin{aligned}\boldsymbol{\lambda} \cdot \boldsymbol{\Phi}(\mathbf{W}, \mathbf{I}) = &\sum_{\mu \in \mathcal{V}} \boldsymbol{\lambda}^1 \cdot \boldsymbol{\phi}^1_\mu(w_\mu, \mathbf{I}) + \boldsymbol{\lambda}^2 \sum_{\mu \in \mathcal{V}} \cdot \boldsymbol{\phi}^2_\mu(w_\mu, \mathbf{I}) \\ &+ \sum_{\mu \in \mathcal{V}/\mathcal{V}^{\text{leaf}}} \boldsymbol{\lambda}^3 \cdot \boldsymbol{\psi}^3_\mu(w_\mu, \mathbf{w}_{ch(\mu)}) \\ &+ \sum_{\mu \in \mathcal{V}/\mathcal{V}^{\text{leaf}}} \boldsymbol{\lambda}^4 \cdot \boldsymbol{\psi}^4_\mu(w_\mu, \mathbf{w}_{ch(\mu)}) \\ &+ \sum_{\mu \in \mathcal{V}} \boldsymbol{\lambda}^5 \cdot \boldsymbol{\psi}^5_\mu(w_\mu) \\ &+ \sum_{\mu \in \mathcal{V}} \boldsymbol{\lambda}^6 \cdot \boldsymbol{\phi}^6_\mu(w_\mu). \end{aligned} \tag{24}$$

The *inference task* is to estimate $\hat{\mathbf{W}} = \arg\max P(\mathbf{W}|\mathbf{I})$ to determine the best interpretation of the input image. The graph structure of the RCM contains no closed loops and hence inference can be done by dynamic programming. Because the state space $w_\mu = (s_\mu, \mathbf{c}_\mu)$ is large we prune to keep a fixed proportion of state based on their energies, see Ref. [44] for details.

The *learning task* is to determine the parameters $\boldsymbol{\lambda}$ from a set of training images with groundtruth $\{(\mathbf{I}_\mu, \mathbf{W}_\mu) : i = 1, \ldots, N\}$. Note that groundtruth for standard datasets is specified only at the leaf nodes, but it is straightforward to approximate the state of the higher levels nodes from this information, see Ref. [44]. We used the structure perceptron algorithm [59]. This is a discriminative learning method which avoid the need to compute the normalization constants $Z(\boldsymbol{\lambda})$ of the probability distributions, unlike standard maximum likelihood methods, and is arguably more effective for discriminative problems.

Formally the algorithm is specified by the update rules:

$$\boldsymbol{\lambda}^{t+1} = \boldsymbol{\lambda}^t - \boldsymbol{\Phi}(\mathbf{W}_\mu, \mathbf{I}_\mu) + \boldsymbol{\Phi}(\mathbf{W}^*(\mathbf{I}_\mu, \boldsymbol{\lambda}), \mathbf{I}_\mu), \tag{25}$$

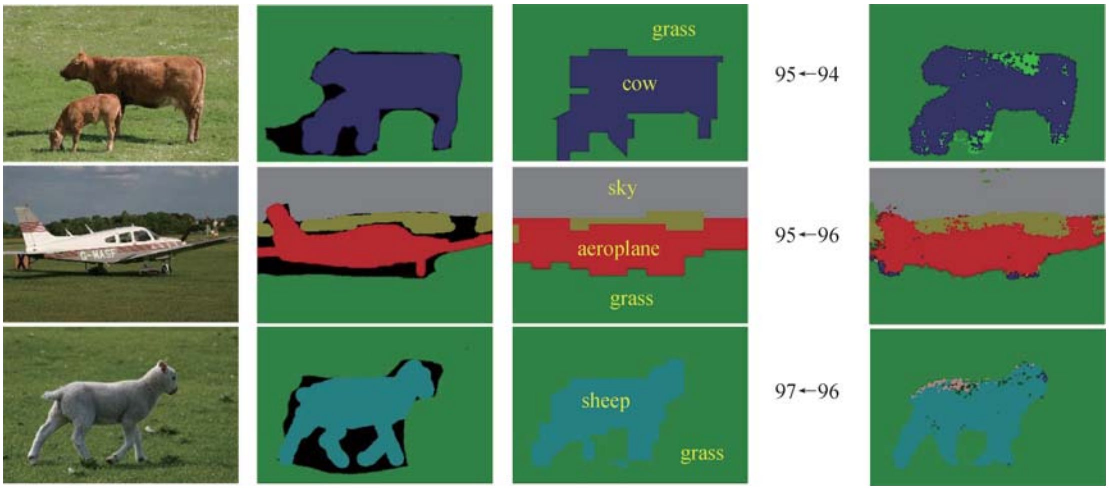


Fig. 6 Parse results on MSRC dataset. The correspondence between the color and the object class is defined as follows. The colors indicate the labels of 21 object classes as in the MSRC dataset [20]. The columns (except the fourth "accuracy" column) show the input images, ground truth, the labels obtained by HIM and the boosting classifier, respectively. The "accuracy" column shows the global accuracy obtained by HIM (left) and the boosting classifier (right)

where i indexes the example from the training dataset which is randomly selected by the algorithm at the iteration time t, and $\mathbf{W}^*(\mathbf{I}_\mu, \boldsymbol{\lambda}) = \arg\min \boldsymbol{\lambda} \cdot \boldsymbol{\Phi}(\mathbf{W}, \mathbf{I}_\mu)$ is the best estimate provided by the inference algorithm (with current settings of $\boldsymbol{\lambda}$).

Structure perceptron has the desirable property, empirically verified for these applications, that many of the weights remain close to zero (with weights initialized at zero). Therefore it acts like a selection process which selects which potentials to use from a dictionary by awarding them large weights.

Datasets and Results

We tested the RCM on the Microsoft dataset [20]. We show example results in Fig. 6 and report good performance compared to state of the art methods, see Ref. [44] for more details.

7 Analyzing models

This section shows alternative ways that concepts from information theory have been used in computer vision. We will concentrate on measures for evaluating and analyzing visual algorithms. This material is drawn from Refs. [32,60–64].

First we introduce the set of models that we will analyze. These are based on deformable models of objects and, to simplify the analysis, we require that the models have no closed loops. Then we will describe how information theoretic measures and techniques can be used to analyze their performance.

7.1 Background: Deformable templates

The models that we analyze in this section are special cases of the deformable template models used to represent objects. Deformable template models have an extensive history in computer vision [65,66]. They can be *represented* in terms of flat probabilistic models [50,67–71] as we now describe. The formulation is again described on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with state variables $\mathbf{W}$ defined on the nodes $\mathcal{V}$, where the state w_μ of node μ represents the position (and possibly orientation) of a point, or part, of the object. The unary potentials $\boldsymbol{\lambda}^D \cdot \boldsymbol{\phi}(w_\mu, \mathbf{I})$ specify how points/parts of the object relate to the image — e.g., some points on the boundary of objects may correspond to edges in the image intensity, while others may be modeled by interest points such as corners [68]. The edges $\mathcal{E}$ specify which points/parts of the object are directly related and the binary potentials $\boldsymbol{\lambda}^P \cdot \boldsymbol{\phi}(w_\mu, w_\nu)$ model the spatial relations — e.g., capturing the overall shape of the object.

This can be expressed by a probability distribution:

$$P(\mathbf{w}|\mathbf{I}) = \frac{1}{Z(\mathbf{I})} \exp\left\{ -\sum_{\mu\in\mathcal{V}} \boldsymbol{\lambda}^D \cdot \boldsymbol{\phi}_\mu(w_\mu, \mathbf{I}) - \sum_{\mu\nu\in\mathcal{E}} \boldsymbol{\lambda}^P \cdot \boldsymbol{\psi}_{\mu\nu}(w_\mu, w_\nu) \right\}, \tag{26}$$

where the main differences are the state variables w_μ at the nodes and the graph structure, see Fig. 7.

Inference is different for these type of models. Firstly, the state variables can typically take a far larger set of values. For example, the set of possible positions in an image is very large. Secondly, the types of graph structure are different. If the object has a chain-like structure — i.e., without closed loops — then dynamic programming can be to perform inference and detect the object independent [68] but pruning is necessary to ensure that the algorithm converges quickly. The computations required by dynamic programming can be sped up using

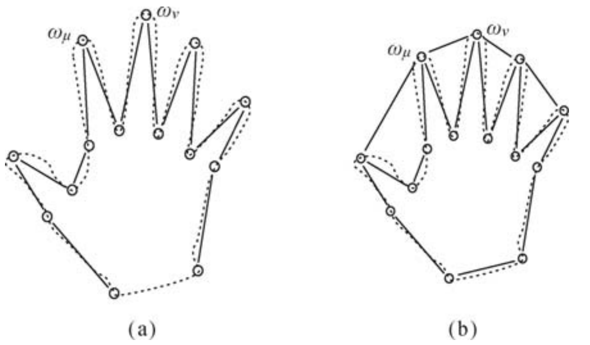


Fig. 7 A deformable template model of a hand without closed loops (a) and with closed loops (b)

various techniques [70]. By choosing more complex image features, such as shape context [50], it is possible to perform good local matches by ignoring the binary terms and then get better matches by use of the Hungarian algorithm. If there are good estimates of the initial configuration, then variational methods can be effective for inference [69]. See Ref. [52] for how shape context and variational methods can be combined effectively. But, once again, the inference algorithms become far less effective if we start introducing longer range edges to capture the spatial regularities of objects.

Learning is possible for these models provided groundtruth data has been specified. It remains straightforward to learn the unary parameters if groundtruth data is specified. Learning the binary parameters is more complicated. In practice, many of the original models were hand specified although stochastic sampling can be performed to validate the model parameters — e.g., to determine if the samples from the object look like real examples of the object [68]. But stochastic sampling is only efficient for certain types of graph structure (e.g., it was straightforward for Coughlan et al. [68] because of their use of models without closed loops). More advanced learning methods are only practical if there is an efficient inference algorithm.

7.2 Analysis and evaluation

7.2.1 Curve model

Suppose we have a model of a deformable shape without closed loops which might, for example, represent a road [67] or the boundary of an object [68]. The model can be formulated as a graph of ordered nodes $\mu \in \mathcal{V}$ with edges $(\mu, \mu+1) \in \mathcal{E}$. The state variables $w_\mu = (\mathbf{W}_\mu)$ can denote position $\vec{x}$ (more and orientation θ. The input is an image $\mathbf{I}$, where $\mathbf{I} = \{I_\mu : \mu \in \mathcal{D}\}$ specifies the intensity values $I_\mu \in \{0, 255\}$ on the lattice. We apply filters (linear or non-linear) to obtain a filtered image $\{z_\mu : \mu \in \mathcal{D}\}$.

We assume a model for generating the data:

$$P(\{z_\mu\}|\{w_\nu\}) = \prod_{\mu \in \{\mathbf{W}_\nu\}} P(z_\mu|s_1) \prod_{\mu \notin \{\mathbf{W}_\nu\}} P(z_\mu|s_2), \tag{27}$$

$$P_G(\{w_\nu\}) = P_G(w_0) \prod_{\mu \in \mathcal{V}} P_G(w_{\nu+1}|w_\nu), \tag{28}$$

where we assume that the features response are drawn from a distribution $P(z_\mu|s_1)$ if point μ lies on the model, and from $P(z_\mu|s_2)$ otherwise. For example, we could consider that the model represents the boundary of the object, set z_μ to be an edge filter (e.g., $z_\mu = |\nabla I(\mu)|$ could be the intensity gradient), and that the distribution will be different for points on the boundary of the object versus points that do not lie on the boundary. For another example, Geman and Jedynak [67] model a road and define a filter which responds preferentially to road segments and which hence the responses on and off a road are drawn from different distributions.

7.2.2 Evaluating cues

Now we analyze how effective the features cues are for detecting the curve ignoring the geometry for the moment. The cues are modeled by $P(z|s_1)$ and $P(z|s_2)$ and we assume priors $P(s_1)$ and $P(s_2)$ for the frequency of the cues (but we ignore the geometry $P(\{w_\nu\})$).

We can evaluate the cues by calculating the Bayes risk for classifying a measurement z as s_1 or s_2. The Bayes decision rule is to decide s_1 if $P(s_1|z) < P(s_2|z)$ and s_2 otherwise. The Bayes risk is given by

$$\begin{aligned} E^* &= \int \min\{p(s_1|z), p(s_2|z)\} p(z) \mathrm{d}z \\ &= \int \min\{p(s_1)p(z|s_1), p(s_2)p(z|s_2)\} \mathrm{d}z. \end{aligned} \tag{29}$$

We can bound this by observing that $\min(a,b) \leqslant a^t b^{1-t}$ for all $t \in [0,1]$, which gives

$$\begin{aligned} E^* &\leqslant p(s_1)^t p(s_2)^{1-t} \exp\{-J_t\}, \text{ with } J_t \\ &= -\log \int p(z|s_1)^t p(z|s_2)^{1-t} \mathrm{d}z. \end{aligned} \tag{30}$$

There are two important special cases obtained by setting $t = 1/2$ and picking $t_C = \arg\max_{t \in [0,1]} J_t$ respectively:

$$\begin{aligned} &E^* \leqslant p(s_1)^{1/2} p(s_2)^{1/2} \mathrm{e}^{-B(p(z|s_1), p(z|s_2))}, \\ &\qquad \text{Bhattacharyya}, \\ &E^* \leqslant p(s_1)^{t_C} p(s_2)^{1-t_C} \mathrm{e}^{-C(p(z|s_1), p(z|s_2))}, \text{ Chernoff}, \end{aligned} \tag{31}$$

where $B(p(z|s_1), p(z|s_2))$ and $C(p(z|s_1), p(z|s_2))$ are the Bhattacharyya bound and the Chernoff information respectively, given by

$$B(p(z|s_1), p(z|s_2)) = -\log \int p(z|s_1)^{1/2} p(z|s_2)^{1/2} \mathrm{d}z,$$

$$\begin{aligned} &C(p(z|s_1), p(z|s_2)) \\ &= -\log \min_{t\in[0,1]} \int p(z|s_1)^t p(z|s_2)^{1-t} \mathrm{d}z \qquad (32) \\ &= -\log \int p(z|s_1)^{t_C} p(z|s_2)^{1-t_C} \mathrm{d}z. \end{aligned}$$

7.2.3 Evaluating groups of cues

Now suppose we have to decide if a group of N pixels are either all from s_1 or all from s_2. We consider two related tasks. The first task is given a set of samples $\{z_i : i = 1, \ldots, N\}$ which are all from $P(z|s_1)$ or from $P(z|s_2)$ and the task is determine whether they comes from s_1 or s_2. This is clearly easier than classifying a single measurement z because we now have N samples. It can be shown, see Refs. [7,72], that the classification error scales as

$$\exp^{-NC(P(\cdot|s_1),P(\cdot|s_2))}, \text{ where } C(P(\cdot|s_1), P(\cdot|s_2)) \text{ is the Chernoff Information.} \qquad (33)$$

The input to the second task is two sets of samples $\{z_1\}$ and $\{z_2\}$ where one is from $P(z|s_1)$ and the other $P(z|s_2)$, but we do not know which. The classification error scales as

$$\exp^{-NB(P(\cdot|s_1),P(\cdot|s_2))}, \text{ where } B(P(\cdot|s_1), P(\cdot|s_2)) \text{ is the Bhattacharyya bound.} \qquad (34)$$

Konishi et al. [72] give many examples of using these measures for classifying the effectiveness of different edge cues.

7.2.4 Detecting a target curve

We next consider the tougher problem of detecting a target curve in an image. This can be formulated using the likelihood and prior specified in Equations (27), (28) and reduces to finding the set of $\{w_\nu\}$ that maximizes the reward function $R(\{w_\nu\}) \propto \log\{P(z_\mu|\{w_\nu\})P_G(\{w_\nu\})\}$, which can be expressed as (ignoring terms independent of $\{w_\nu\}$)

$$\begin{aligned} R(\{w_\nu\}) = &\sum_{\mu\in\{\mathbf{W}_\nu\}} \log \frac{P(z_\mu|s_1)}{P(z_\mu|s_2)} + \log P(w_0) \\ &+ \sum_{\mu\in\mathcal{V}} \log P_G(w_{\nu+1}|w_\nu). \qquad (35) \end{aligned}$$

For simplicity, we drop the $\log P(w_0)$ term (either by assuming that the initial position w_0 is known — as for Geman and Jedynak [67] or it is specified by a uniform distribution as in Ref. [68].

To determine the difficulty of the detection task, we can first evaluate the expected reward if we correctly detect the position of the curve. This assumes that the data $\{z_\nu\}$ are sampled from the distribution $P(z|s_1)$ and the positions $\{w_\nu\}$ are sampled from $P_G(w_{\nu+1}|w_\nu)$. This can be compared to the expected reward for curves which lie off the true position of the curve. Their data values $\{z_\nu\}$ are sampled from $P(z|s_2)$ and their positions $\{w_\nu\}$ are sampled from a default (e.g., uniform) distribution $U(w_{\nu+1}|w_\nu)$. The expected rewards for both cases are given by

$$\begin{aligned} \langle R(\{w_\nu\})\rangle_{s_1} &= ND(P(\cdot|s_1), P(\cdot|s_2)) - NH(P_G), \\ \langle R(\{w_\nu\})\rangle_{s_1} &= -ND(P(\cdot|s_2), P(\cdot|s_1)) \qquad (36) \\ &\quad +N\sum_w U(w)\log P_G(w). \end{aligned}$$

From this we see that expected reward on the true position of the curve is much bigger than the expected reward of any curve in this image. The difference is $N\{D(P(\cdot|s_1), P(\cdot|s_2)) + D(P(\cdot|s_2), P(\cdot|s_1)) + D(P_G|U) + D(U|P_G)\}$. But this analysis ignores the fact that there are exponentially 'false curves' in the image and only one target curve.

7.2.5 Order parameters

A series of papers [60–64] analyzed the difficulty of detecting curves in images taking into account the exponential number of false targets. The analysis used Sanov Theorem [7] and related techniques to estimate the probability of rare events — i.e., that a "false curve" would have larger reward than the target curve. The analysis made certain simplifying assumptions (e.g., the target curves do not intersect) which were checked by computer simulations [64].

The analysis showed that the performance depended on an *order parameter* K provided the length N of the target curve was sufficiently large. The value of K is given by

$$K = D(P(\cdot|s_1), P(\cdot|s_2)) + D(P_G|U) - H(U), \qquad (37)$$

where $H(U)$ is the entropy of the uniform distribution.

If $K < 0$ then it will be impossible to detect the target curve since, with high probability, there will be a false curve whose reward is higher than that of the target curve. In fact, the Bayes risk for detecting the target curve will tend to 0 as $N \mapsto \infty$ only if $K > 0$, see Ref. [64].

Further analysis [60] shows that the speed of A* inference algorithms for detecting the target curve (provided it is detectable) will also depend on the order parameter K. The intuition is simple — the amount of time that the inference algorithm wastes while searching for the target curve depends on how easy it is to confuse the target curve with random curves in the background, which depends on K. Overall, the expected convergence rate of A* and related algorithms is $O(N)$ with a constant that depends on K (and which decreases as K gets increasingly negative).

7.2.6 High-low: Value of information

The analysis above assumed that we performed inference using the models that generated the data. But what happens if we try to perform inference using a simpler approximate model? This situation can happen because, of realistic problems, we will not know the 'true model' but can only approximate it. In addition, there may be computational advantages to using a simpler model for inference because it may give a more efficient algorithm.

We now analyze the value of information *lost* by using a weaker prior model. More precisely, in place of the correct geometric model $P_G(\cdot)$ we replace it by a weaker model $P_H(\cdot)$. This defines two different rewards R_G and R_H:

$$\begin{aligned} R_G(\{w_\mu\}) &= \sum_\mu \log \frac{P(z_\mu|s_1)}{P(z_\mu|s_2)} + \sum_\mu \log \frac{P_G(w_\mu)}{U(w_\mu)}, \\ R_H(\{w_\mu\}) &= \sum_\mu \log \frac{P(z_\mu|s_1)}{P(z_\mu|s_2)} + \sum_\mu \log \frac{P_H(w_\mu)}{U(w_\mu)}. \end{aligned} \tag{38}$$

The optimal Bayesian strategy to search for the road would be to use the correct model $P_G(\cdot)$ and hence evaluate curves based on the reward R_G. But how much do we lose by using the weaker model $P_H(\cdot)$ and reward R_H? There may be significant computational savings by performing inference using a weak first order Markov model P_H when the real prior is a second order Markov model P_G.

The earlier order parameter analysis can be extended to deal with the case when we perform inference using the weaker model [63,64]. We compute an order parameter K_H for the weaker model which are, by necessity, smaller than the order parameter K of the correct model. Since the Bayes risk depends on K (in the limit of large N) this means that there will situations where we the weaker model is able to detect the target curve but also cases where the correct model can detect the target while the weak model cannot:

$$\begin{aligned} &K > K_H > 0, && \text{weak model sufficient for detection,} \\ &K > 0 > K_H, && \text{correct model required,} \\ &0 > K > K_H, && \text{impossible to detect target by} \\ & && \text{any model.} \end{aligned} \tag{39}$$

There is a particularly important special case which arises if P_H satisfies the condition:

$$\sum_{\mathbf{W}} P_G(\mathbf{W}) \log P_H(\mathbf{W}) = \sum_{\mathbf{W}} P_H(\mathbf{W}) \log P_H(\mathbf{W}). \tag{40}$$

We call this the *Amari* condition because it arises within Amari's theory of information geometry [2]. It corresponds to P_H being the projection of P_G onto an exponential sub-manifold. This relates to minimax entropy theory as follows. Suppose we learn a distribution $P_H(\mathbf{W}) = \frac{1}{Z[\boldsymbol{\lambda}_1]} \exp\{\boldsymbol{\lambda}_1 \cdot \boldsymbol{\phi}_1(\mathbf{W})\}$ where $\sum_{\mathbf{W}} \boldsymbol{\phi}_1(\mathbf{W}) P_H(\mathbf{W}) = \boldsymbol{\psi}_1$. Now consider a distribution $P_G(\mathbf{W})$ which depends on two sufficient potentials $\boldsymbol{\phi}_1(\mathbf{W})$ and $\boldsymbol{\phi}_2(\mathbf{W})$ — i.e., $P_G(\mathbf{W}) = \frac{1}{Z[\boldsymbol{\nu}_1, \boldsymbol{\nu}_2]} \exp\{\boldsymbol{\nu}_1 \cdot \boldsymbol{\phi}_1(\mathbf{W}) + \boldsymbol{\nu}_2 \cdot \boldsymbol{\phi}_2(\mathbf{W})\}$ — with condition that $\sum_{\mathbf{W}} \boldsymbol{\phi}_1(\mathbf{W}) P_G(\mathbf{W}) = \boldsymbol{\psi}_1$ and $\sum_{\mathbf{W}} \boldsymbol{\phi}_2(\mathbf{W}) P_G(\mathbf{W}) = \boldsymbol{\psi}_2$. It can easily be checked that the Amari condition, Equation (40), is satisfied. Hence the Amari condition will be satisfied if P_H is obtained using the maximum entropy principle using a subset of the constraints used to determine P_G.

In this special case, the order parameter increases by $D(P_H||P_G)$ if we use the weaker model P_H for inference. This helps clarify the value of using additional potentials in minimax entropy learning.

8 Discussion

This paper has given an introduction to computer vision from an information theory perspective. We have described probabilistic models for images and how they can be used for encoding. We have stressed the necessity of having efficient inference and learning algorithms for these models and emphasized approaches that use concepts from information theory. We emphasize that natural images are extremely complex and that progress in computer vision comes from having richer and more realistic image models. In particular, the classical MRF models of images are being replaced by more advanced models that involve stochastic grammars and hierarchies. Finally, we described how techniques from information theory can be used to analyze vision models and measure the effectiveness of different visual cues.

We stress that this article has only given a brief introduction to this exciting area on the boundary of computer vision and information theory. A recent book [5] gives far more details about many of the topics covered here and describes others that we have not had space for. For example, mutual information [73] has been used as a measure for finding the correspondence between two images. This is particularly useful for applications in medical images where the task is to match images that were taken under different parameter settings which causes non-linear transformations to the images. The intensities of corresponding points in the images may differ greatly but mutual information is largely invariant to these transformations and so can yield good matching. Recent work has lead to greater understanding of why this measure is so effective [74].

Finally, it has long been argued [75,76] that vision is an active process which involves exploring the real world. Visual systems — human or robotic — can move in the world to obtain additional information. Soatto [77] has recently argued that this requires modifying and enhancing information theoretic concepts in order to deal with this dynamic aspect of vision.

Acknowledgements The author would like to acknowledge funding support from NSF with grants IIS-0917141 and 0613563 and from AFOSR FA9550-08-1-0489.

References

1. Barlow H B. The absolute efficiency of perceptual decisions. Philosophical Transactions of the Royal Society of London (Series B), 1980, 290(1038): 71–82
2. Amari S. Differential geometry of curved exponential families — curvature and information loss. Annals of Statistics, 1982, 10(2): 357–385
3. Amari S. Information geometry and its applications: Convex function and dually flat manifold. In: Proceedings of Emerging Trends in Visual Computing. Lecture Notes in Computer Science, 2009, 5416: 75–102
4. Xu L. Bayesian Ying-Yang machine, clustering and number of clusters. Pattern Recognition Letters, 1997, 18(11–13): 1167–1178
5. Escolano F, Suau P, Bonev B. Information Theory in Computer Vision and Pattern Recognition. Springer, 2009
6. Shannon C E. A mathematical theory of communication. Bell System Technical Journal, 1948, 27: 379–423, 623–656
7. Cover T M, Thomas J A. Elements of Information Theory. New York: Wiley-Interscience, 1991
8. Kanizsa G. Organization in Vision. New York: Praeger, 1979
9. Gregory R L. The Intelligent Eye. London: Weidenfeld and Nicolson, 1970
10. Lee T S, Mumford D. Hierarchical Bayesian inference in the visual cortex. Journal of the Optical Society of America A, 2003, 20(7): 1434–1448
11. Atick J J, Redlich A N. What does the retina know about natural scenes? Neural Computation, 1992, 4(2): 196–210
12. Olshausen B A, Field D J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature, 1996, 381(6583): 607–609
13. Grenander U. General Pattern Theory. Oxford University Press, 1993
14. IPAM Summer School: The mathematics of the mind. Tenenbaum J B, Yuille A L, Organizers. IPAM, UCLA. 2007
15. Jin Y, Geman S. Context and hierarchy in a probabilistic image model. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2006, 2: 2145–2152
16. Zhu S C, Mumford D. A stochastic grammar of images. Foundations and Trends in Computer Graphics and Vision, 2006, 2(4): 259–362
17. Leclerc Y G. Constructing simple stable descriptions for image partitioning. International Journal of Computer Vision, 1989, 3(1): 73–102
18. Rissanen J. Minimum description length principle. In: Kotz S, Johnson N L, eds. Encyclopedia of Statistical Sciences. New York: John Wiley & Sons, 1987, 5: 523–527
19. Geman S, Geman D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1984, PAMI-6(6): 721–741
20. Shotton J, Winn J, Rother C, Criminisi A. TextonBoost: Joint appearance, shape and context modeling for multiclass object recognition and segmentation. In: Proceedings of the 9th European Conference on Computer Vision. Lecture notes in computer science, 2006, 3951: 1–15
21. Geiger D, Ladendorf B, Yuille A L. Occlusions and binocular stereo. International Journal of Computer Vision, 1995, 14(3): 211–226
22. Sun J, Shum H-Y, Zheng N-N. Stereo matching using belief propagation. In: Proceedings of the 7th European Conference on Computer Vision. Lecture notes in computer science, 2002, 2351: 510–524
23. Blake A, Zisserman A. Visual Reconstruction. Cambridge: MIT Press, 1987
24. Geiger D, Yuille A L. A common framework for image segmentation. International Journal of Computer Vision, 1991, 6(3): 227–243
25. Black M J, Rangarajan A. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. International Journal of Computer Vision, 1996, 19(1): 57–91
26. Zhu S C, Mumford D. Prior learning and Gibbs reaction diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(11): 1236–1250
27. Roth S, Black M J. Fields of experts. International Journal of Computer Vision, 2009, 82(2): 205–229
28. Boykov Y, Kolmogorov V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In: proceedings of the Energy Minimization Methods in Computer Vision and Pattern Recognition. Lecture Notes in Computer Science, 2001, 2134: 359–374
29. Koch C, Marroquin J, Yuille A L. Analog "neuronal" networks in early vision. Proceedings of the National Academy of Sciences of the United States of America, 1986, 83(12): 4263–4267
30. Yedidia J S, Freeman W T, Weiss Y. Generalized belief propagation. Advances in Neural Information Processing Systems, 2001, 13: 689–695
31. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Conference on Computer Vision and Pattern Recognition. 2001, 1: I-511–I-518
32. Konishi S, Yuille A L, Coughlan J M, Zhu S C. Fundamental bounds on edge detection: An information theoretic evaluation of different edge cues. In: Proceedings of the International Conference on Computer Vision and Pattern Recognition. 1999, 1: 573–579
33. Lafferty J, McCallum A, Pereira F. Conditional random

fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning. 2001, 282–289

34. Zhu S C, Wu Y N, Mumford D. Minimax entropy principle and its application to texture modeling. Neural Computation, 1997, 9(8): 1627–1660
35. Parisi G. Statistical Field Theory. Addison Wesley, 1988
36. Hopfield J J, Tank D W. "Neural" computation of decisions in optimization problems. Biological Cybernetics, 1985, 52(3): 141–152
37. Saul L, Jordan M. Exploiting tractable substructures in intractable networks. Advances in Neural Information Processing Systems, 1995, 8: 486–492
38. Wainwright M J, Jaakkola T S, Willsky A S. Tree-based reparameterization framework for analysis of sum-product and related algorithms. IEEE Transactions on Information Theory, 2003, 49(5): 1120–1146
39. Bishop C M. Pattern Recognition and Machine Learning. 2nd ed. Springer, 2007
40. Domb C, Green M S. Phase Transitions and Critical Phenomena. London: Academic Press, 1972
41. Neal R M, Hinton G E. A view of the EM Algorithm that justifies incremental, sparse, and other variants. In: Jordan M I ed. Learning in Graphical Models. Cambridge: MIT Press, 1999, 355–368
42. Tu Z, Zhu S C. Image segmentation by data-driven Markov chain Monte Carlo. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(5): 657–673
43. Tu Z W, Chen X, Yuille A L, Zhu S C. Image parsing: Unifying segmentation, detection, and recognition. International Journal of Computer Vision, 2005, 63(2): 113–140
44. Zhu L, Chen Y, Lin Y, Yuille A L. A hierarchical image model for polynomial-time 2D parsing. In: Proceedings of Neural Information Processing Systems Foundation. 2008
45. Zhu S C, Yuille A L. Region competition: Unifying snakes, region growing and Bayes/MDL for multiband image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, 18(9): 884–900
46. Gilks W R, Richardson S, Spiegelhalter D J. Markov Chain Monte Carlo in Practice. Chapman & Hall, 1996
47. Freund Y, Schapire R. Experiments with a new boosting algorithm. In: Proceedings of the Thirteenth International Conference on Machine Learning. 1996, 148–156
48. Chen X, Yuille A L. A time-efficient cascade for real-time object detection: With applications for the visually impaired. In: Proceedings of Computer Vision and Pattern Recognition. 2005, 28
49. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning. 2nd ed. Springer, 2009
50. Belongie S, Malik J, Puzicha J. Matching shapes. In: Proceedings of the Eighth IEEE International Conference on Computer Vision, 2001, 1: 454–461
51. Cootes T F, Edwards G J, Taylor C J. Active appearance models. In: proceedings of the 5th European Conference on Computer Vision. Lecture Notes in Computer Science, 1998, 1407: 484–498
52. Tu Z, Yuille A L. Shape matching and recognition: Using generative models and informative features. In: Proceedings of the 8th European Conference on Computer Vision. 2004, 3: 195–209
53. Martin D, Fowlkes C, Tal D, Malik J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of the Eighth International Conference on Computer Vision. 2001, 2: 416–423
54. Guo C E, Zhu S C, Wu Y N. Primal sketch: Integrating structure and texture. Computer Vision and Image Understanding, 2007, 106(1): 5–19
55. Chen H, Xu Z, Liu Z, Zhu S C. Composite templates for cloth modeling and sketching. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2006, 1: 943–950
56. Zhu L, Lin C, Huang H, Chen Y, Yuille A L. Unsupervised structure learning: Hierarchical recursive composition, suspicious coincidence and competitive exclusion. In: Proceedings of the 10th European Conference on Computer Vision. Lecture Notes in Computer Science, 2008, 5303: 759–773
57. Zhu L, Chen Y, Lu Y, Lin C, Yuille A L. Max margin AND/OR graph learning for parsing the human body. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2008
58. Zhu L, Chen Y, Ye X, Yuille A L. Structure-perceptron learning of a hierarchical log-linear model. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2008
59. Collins M. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. 2002, 1–8
60. Coughlan J M, Yuille A L. Bayesian A* tree search with expected O(N) convergence rates for road tracking. In: proceedings of Energy Minimization Methods in Computer Vision and Pattern Recognition. Lecture Notes in Computer Science, 1999, 1654: 189–204
61. Yuille A L, Coughlan J M. Fundamental limits of Bayesian inference: Order parameters and phase transitions for road tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(2): 160–173
62. Yuille A L, Coughlan J M. An A* perspective on deterministic optimization for deformable templates. Pattern Recognition, 2000, 33(4): 603–616
63. Yuille A L, Coughlan J M. High-level and generic models for visual search: When does high level knowledge help? In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 1999, 2: 631–637
64. Yuille A L, Coughlan J M, Wu Y N, Zhu S C. Order parameters for detecting target curves in images: When does high level knowledge help? International Journal of Computer Vision, 2001, 41(1–2): 9–33
65. Fischler M A, Elschlager R A. The representation and matching of pictorial structures. IEEE Transactions on Computers, 1973, C-22(1): 67–92
66. Yuille A L, Hallinan P W, Cohen D S. Feature extraction from faces using deformable templates. International Journal of Computer Vision, 1992, 8(2): 99–111
67. Geman D, Jedynak B. An active testing model for tracking roads in satellite images. IEEE Transactions on Pattern

Analysis and Machine Intelligence, 1996, 18(1): 1–14

68. Coughlan J M, Yuille A L, English C, Snow D. Efficient deformable template detection and localization without user initialization. Computer Vision and Image Understanding, 2000, 78(3): 303–319
69. Chui H, Rangarajan A. A new algorithm for non-rigid point matching. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2000, 2: 44–51
70. Felzenszwalb P F, Huttenlocher D P. Pictorial structures for object recognition. International Journal of Computer Vision, 2005, 61(1): 55–79
71. Fergus R, Perona P, Zisserman A. A sparse object category model for efficient learning and exhaustive recognition. In: proceeding of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2005, 1: 380–387
72. Konishi S, Yuille A L, Coughlan J M, Zhu S C. Statistical edge detection: Learning and evaluating edge cues. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(1): 57–74
73. Viola P, Wells W M III. Alignment by maximization of mutual information. International Journal of Computer Vision, 1997, 24(2): 137–154
74. Rajwade A, Banerjee A, Rangarajan A. Probability density estimation using isocontours and isosurfaces: Applications to information-theoretic image registration. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(3): 475–491
75. Gibson J J. The Ecological Approach to Visual Perception. Boston: Houghton Mifflin, 1979
76. Blake A, Yuille A L. Active Vision. Cambridge: MIT Press, 1992
77. Soatto S. Actionable information in vision. In: Proceedings of the International Conference on Computer Vision. 2009, 2425

Alan Yuille received his B.A. in mathematics from the University of Cambridge in 1976, and completed his Ph.D. in theoretical physics at Cambridge in 1980 studying under Stephen Hawking. Following this, he held a postdoc position with the Physics Department, University of Texas at Austin, and the Institute for Theoretical Physics, Santa Barbara. He then joined the Artificial Intelligence Laboratory at MIT (1982–1986), and followed this with a faculty position in the Division of Applied Sciences at Harvard (1986–1995), rising to the position of associate professor. From 1995–2002 Alan worked as a senior scientist at the Smith-Kettlewell Eye Research Institute in San Francisco. In 2002 he accepted a position as full professor in the Department of Statistics at the University of California, Los Angeles. He has over one hundred and fifty peer-reviewed publications in vision, neural networks, and physics, and has co-authored two books: *Data Fusion for Sensory Information Processing Systems* (with J. J. Clark) and *Two- and Three-Dimensional Patterns of the Face* (with P. W. Hallinan, G. G. Gordon, P. J. Giblin and D. B. Mumford); he also co-edited the book *Active Vision* (with A. Blake). He has won several academic prizes and is a Fellow of IEEE.