Chapter 8

Parameter Estimation in Optimal Object Recognition

Object recognition systems involve parameters such as thresholds, bounds, and weights. These parameters have to be tuned before the system can perform successfully. A common practice is to choose such parameters manually on an ad hoc basis, which is a disadvantage. This chapter[1] presents a theory of parameter estimation for optimization-based object recognition where the optimal solution is defined as the global minimum of an energy function. The theory is based on supervised learning from training examples. *Correctness* and *instability* are established as criteria for evaluating the estimated parameters. A correct estimate enables the labeling implied in each example configuration to be encoded in a unique global energy minimum. The instability is the ease with which the minimum is replaced by a nonexample configuration after a perturbation. The optimal estimate minimizes the instability. Algorithms are presented for computing correct and minimal-instability estimates. The theory is applied to the parameter estimation for MRF-based recognition, and promising results are obtained.

8.1 Motivation

Object recognition systems almost inevitably involve parameters such as thresholds, bounds, and weights (Grimson 1990). In optimization-based object recognition, where the optimal recognition solution is explicitly defined as the global extreme of an objective function, these parameters can be part of the definition of the objective function by which the global cost (or gain)

[1] This chapter is based on Li (1997b).

S.Z. Li, *Markov Random Field Modeling in Image Analysis*,
Advances in Pattern Recognition, DOI: 10.1007/978-1-84800-279-1_8,

of the solution is measured. The selection of the parameters is crucial for a system to perform successfully.

Among all the admissible parameter estimates, only a subset of them lead to the desirable or correct solutions to the recognition. Among all the correct estimates, a smaller number of them are better in the sense that they lead to correct solutions for a larger variety of data sets. One of them may be optimal in the sense that it makes the vision procedure the most stable to uncertainties and the least prone to local optima in the search for the global optimum.

The manual method performs parameter estimation in an ad hoc way by trial and error: A combination of parameters is selected to optimize the objective function, and then the optimum is compared with the desirable result in the designer's perception and the selection is adjusted. This process is repeated until a satisfactory choice, that makes the optimum consistent with the desirable result, is found. This is a process of *supervised learning from examples*. When the objective function takes a right functional form, a correct manual selection may be made for a small number of data sets. However, there is no reason to believe that the manual selection is an optimal or even a good one. Such empirical methods have been criticized for their ad hoc nature.

This chapter aims to develop an automated optimal approach for parameter estimation[2] applied to optimization-based object recognition schemes. A theory of parameter estimation based on supervised learning is presented. The learning is "supervised" because a training set of examples is given. Each example represents a desirable recognition result where a recognition result is a labeling of the scene in terms of the model objects. *Correctness* and *optimality* are proposed as the two-level criteria for evaluating parameters estimates.

A correct selection of parameters enables the configuration given by each example to be embedded as a unique global energy minimum. In other words, if the selection is incorrect, the example configuration will not correspond to a global minimum. While a correct estimate can be learned from the examples, it is generally not the only correct solution. *Instability* is defined as the measure of the ease with which the global minimum is replaced by a nonexample labeling after a perturbation to the input. The optimality minimizes the *instability* so as to maximize the ability to generalize the estimated parameters to other situations not directly represented by the examples.

Combining the two criteria gives a constrained minimization problem: minimize the instability subject to the correctness. A nonparametric algorithm is presented for learning an estimate which is optimal as well as correct. It does not make any assumption about the distributions and is useful for cases where the size of the training example set is small and where the

[2] In this chapter, parameter "selection", "estimation" and "learning" are used interchangeably.

underlying parametric models are not accurate. The estimate thus obtained is optimal w.r.t. the training data.

The theory is applied to a specific model of MRF recognition proposed in (Li 1994a). The objective function in this model is the posterior energy of an MRF. The form of the energy function has been derived, but it involves parameters that have to be estimated. The optimal recognition solution is the maximum a posteriori (MAP) configuration of an MRF. Experiments conducted show very promising results in which the optimal estimate serves well for recognizing other scenes and objects.

A parametric method based on *maximum likelihood* is also described for computing the optimal parameter estimate under the Gaussian-MRF assumption. It takes advantage of the assumption and may be useful when the size of the training data is sufficiently large. The parameter estimate thus computed is optimal w.r.t. the assumption.

Although automated and optimal parameter selection for object recognition in high-level vision is an important and interesting problem that has existed for a long time, reports on this topic are rare. Works have been done in related areas. In (Poggio and Edelman 1990), to recognize 3D objects from different viewpoints, a function mapping any viewpoint to a standard view is learned from a set of perspective views. In (Weng et al. 1993), a network structure is introduced for automated learning to recognize 3D objects. In (Pope and Lowe 1993), a numerical graph representation for an object model is learned from features computed from training images. In (Pelillo and Refice 1994) a procedure is proposed for learning compatibility coefficients for relaxation labeling by minimizing a quadratic error function. Automated and optimal parameter estimation for low-Level problems has achieved significant progress. MRF parameter selection has been dealt with in statistics (Besag 1974; Besag 1975) and in applications such as image restoration, reconstruction, and texture analysis (Cross and Jain 1983; Cohen and Cooper 1987; Derin and Elliott 1987; Qian and Titterington 1989; Zhang 1988; Nadabar and Jain 1992). The problem is also addressed from the regularization viewpoint (Wahba 1980; Geiger and Poggio 1987; Shahraray and Anderson 1989; Thompson et al. 1991).

The chapter is organized as follows. Section 8.2 presents the theory. Section 8.3 applies the theory to an MRF recognition model. Section 8.4 presents the experimental results. Finally, conclusions are made in Section 8.5.

8.2 Theory of Parameter Estimation for Recognition

In this section, correctness, instability, and optimality are proposed for evaluating parameter estimates. Their relationships to nonparametric pattern recognition are discussed, and nonparametric methods for computing correct and optimal estimates are presented. Before preceeding, necessary notations for optimization-based object recognition are introduced.

8.2.1 Optimization-Based Object Recognition

In optimization-based recognition, the optimal solution is explicitly defined as the extreme of an objective function. Let f be a configuration representing a recognition solution. The cost of f is measured by a global objective function $E(f \mid \theta)$, also called the energy. The definition of E is dependent on f and a number of $K+1$ parameters $\theta = [\theta_0, \theta_1, \ldots, \theta_K]^T$. As the optimality criterion for model-based recognition, it also relates to other factors such as the observation, denoted $\mathcal{G}$, and model references, denoted $\mathcal{G}'$. Given $\mathcal{G}$, $\mathcal{G}'$, and θ, the energy maps a solution f to a real number by which the cost of the solution is evaluated. The optimal solution corresponds to the global energy minimum, expressed as

$$f^* = \arg\min_f E(f \mid \theta) \tag{8.1}$$

In this regard, it is important to formulate the energy function so that the "correct solution" is embedded as the global minimum. The energy may also serve as a guide to the search for a minimal solution. In this respect, it is desirable that the energy should differentiate the global minimum from other configurations as much as possible.

The energy function may be derived using one of the following probabilistic approaches: fully parametric, partially parametric, and nonparametric. In the fully parametric approach, the energy function is derived from probability distributions in which all the parameters involved are known. Parameter estimation is a problem only in the partially parametric and nonparametric cases.

In the partially parametric case, the forms of distributions are given but some parameters involved are unknown. One example is the Gaussian distribution with an unknown variance, and another is the Gibbs distribution with unknown clique potential parameters. In this case, the problem of estimating parameters in the objective function is related to estimating parameters in the related probability distributions.

In the nonparametric approach, no assumptions are made about distributions and the form of the objective function is obtained based on experiences or prespecified "basis functions" (Poggio and Edelman 1990). This also applies to situations where the data set is too small to have statistical significance.

An important form for E in object recognition is the weighted sum of various terms, expressed as

$$E(f \mid \theta) = \theta^T U(f) = \sum_{k=0}^{K} \theta_k U_k(f) \tag{8.2}$$

where $U(f) = [U_0(f), U_1(f), \ldots, U_K(f)]^T$ is a vector of *potential functions*. A potential function is dependent on f, $\mathcal{G}$, and $\mathcal{G}'$, where the dependence can be nonlinear in f, and often measures the violation of a certain constraint incurred by the solution f. This linear combination of (nonlinear) potential functions is not an unusual form. It has been used in many matching and recognition works; see (Duda and Hart 1973; Fischler and Elschlager 1973; Davis 1979; Ghahraman et al. 1980; Jacobus et al. 1980; Shapiro and Haralick 1981; Oshima and Shirai 1983; Bhanu and Faugeras 1984; Wong and You 1985; Fan et al. 1989; Nasrabadi et al. 1990; Wells 1991; Weng et al. 1992; Li 1994a). Note that when $E(f \mid \theta)$ takes the linear form, multiplying θ by a positive factor $\kappa > 0$ does not change the minimal configuration

$$\arg\min_f E(f \mid \theta) = \arg\min_f E(f \mid \kappa\theta) \tag{8.3}$$

Because of this equivalent, an additional constraint should be imposed on θ for the uniqueness. In this work, θ is confined to having a unit Euclidean length

$$\|\theta\| = \sqrt{\sum_{k=0}^{K} \theta_k^2} = 1 \tag{8.4}$$

Given an observation $\mathcal{G}$, a model reference $\mathcal{G}'$, and the form of $E(f \mid \theta)$, it is the θ value that completely specifies the energy function $E(f \mid \theta)$ and thereby defines the minimal solution f^*. It is desirable to learn the parameters from examples so that the minimization-based recognition is performed correctly. The criteria for this purpose are established in the next subsection.

8.2.2 Criteria for Parameter Estimation

An example is specified by a triple $(\bar{f}, \mathcal{G}, \mathcal{G}')$, where $\bar{f}$ is the example configuration (recognition solution) telling how the scene ($\mathcal{G}$) should be labeled or interpreted in terms of the model reference ($\mathcal{G}'$). The configuration $\bar{f}$ may be a structural mapping from $\mathcal{G}$ to $\mathcal{G}'$. Assume that there are L model objects; then at least L examples have to be used for learning to recognize the L object. Let the instances be given as

$$\{(\bar{f}^\ell, \mathcal{G}^\ell, \mathcal{G}'^\ell) \mid \ell = 1, \ldots, L\} \tag{8.5}$$

We propose two-level criteria for learning θ from examples:

1. *Correctness.* This defines a parameter estimate that encodes constraints into the energy function in a correct way. A correct estimate, denoted $\theta_{correct}$, should embed each $\bar{f}^\ell$ into the minimum of the corresponding energy $E(f^\ell \mid \theta_{correct})$, that is

$$\bar{f}^\ell = \arg\min_f E^\ell(f \mid \theta_{correct}) \qquad \forall \ell \tag{8.6}$$

where the definition of $E^\ell(f \mid \theta)$ is dependent on the given scene $\mathcal{G}^\ell$ and the model $\mathcal{G}'^\ell$ of a particular example $(\bar{f}^\ell, \mathcal{G}^\ell, \mathcal{G}'^\ell)$ as well as θ. Briefly, a correct θ is one that makes the minimal configuration f^* defined in (8.1) coincide with the example configuration $\bar{f}^\ell$.

2. *Optimality.* This is aimed at maximizing the generalizability of the parameter estimate to other situations. A measure of *instability* is defined for the optimality and is to be minimized.

The correctness criterion is necessary for a vision system to perform correctly and is of fundamental importance. Only when this is met does the MRF model make correct use of the constraints. The optimality criterion is not necessary in this regard but makes the estimate most generalizable.

Correctness

In (8.6), a correct estimate $\theta_{correct}$ enables the $\bar{f}$ to be encoded as the global energy minimum for this particular pair $(\mathcal{G}, \mathcal{G}')$. Therefore, it makes any energy change due to a configuration change from $\bar{f}$ to $f \neq \bar{f}$ positive; that is,

$$\begin{aligned}\Delta E^\ell(f \mid \theta) &= E^\ell(f \mid \theta) - E^\ell(\bar{f}^\ell \mid \theta) \\ &= \sum_k \theta_k [U_k(f) - U_k(\bar{f}^\ell)] > 0 \qquad \forall f \in \mathcal{F}^\ell\end{aligned} \tag{8.7}$$

where

$$\mathcal{F}^\ell = \{f \mid f \neq \bar{f}^\ell\} \tag{8.8}$$

is the set of all non-$\bar{f}^\ell$ configurations. Let

$$\Theta_{correct} = \{\theta \mid \Delta E^\ell(f \mid \theta) > 0, \forall f \in \mathcal{F}^\ell, \ell = 1, \ldots, L\} \tag{8.9}$$

be the set of all correct estimates. The set, if non-empty, usually comprises not just a single point but a region in the allowable parameter space. Some of the points in the region are better in the sense of stability, to be defined below.

Instability

The value of the energy change $\Delta E^\ell(f \mid \theta) > 0$ can be used to measure the (local) stability of $\theta \in \Theta_{correct}$ w.r.t. a certain configuration $f \in \mathcal{F}^\ell$. Ideally, we want $E^\ell(\bar{f}^\ell \mid \theta)$ to be very low and $E^\ell(f \mid \theta)$ to be very high, such that $\Delta E^\ell(f \mid \theta)$ is very large, for all $f \neq \bar{f}^\ell$. In such a situation, $\bar{f}^\ell$ is expected to be a stable minimum where the stability is w.r.t. perturbations in the observation and w.r.t. the local minimum problem with the minimization algorithm.

The smaller $\Delta E^\ell(f \mid \theta)$ is, the larger is the chance with which a perturbation to the observation will cause $\Delta E^\ell(f \mid \theta_{correct})$ to become negative to violate the correctness. When $\Delta E^\ell(f \mid \theta_{correct}) < 0$, $\bar{f}^\ell$ no longer corresponds to the global minimum. Moreover, we assume that configurations f whose energies are slightly higher than $E^\ell(\bar{f}^\ell \mid \theta)$ are possibly *local energy minima* at which an energy minimization algorithm is most likely to get stuck.

Therefore, the energy difference (i.e., the local stabilities) should be enlarged. One may define the global stability as the sum of all $\Delta E^\ell(f \mid \theta)$. For reasons to be explained later, *instability*, instead of stability, is used for evaluating θ.

The local instability for a correct estimate $\theta \in \Theta_{correct}$ is defined as

$$c^\ell(\theta, f) = \frac{1}{\Delta E^\ell(f \mid \theta)} \tag{8.10}$$

where $f \in \mathcal{F}^\ell$. It is "local" because it considers only one $f \in \mathcal{F}^\ell$. It is desirable to choose θ such that the value of $c^\ell(\theta, f)$ is small *for all* $f \in \mathcal{F}^\ell$. Therefore, we defined the global *p-instability* of θ

$$C_p^\ell(\theta) = \left\{ \sum_{f \in \mathcal{F}^\ell} \left[c^\ell(\theta, f)\right]^p \right\}^{1/p} \tag{8.11}$$

where $p \geq 1$. The total global p-instability of θ is

$$C_p(\theta) = \sum_{\ell=1}^{L} C_p^\ell(\theta) \tag{8.12}$$

In the limit as $p \to \infty$, we have that[3]

$$C_\infty^\ell(\theta) = \max_{f \in \mathcal{F}^\ell} c^\ell(\theta, f) = \frac{1}{\min_{f \in \mathcal{F}^\ell} \Delta E^\ell(f \mid \theta)} \tag{8.13}$$

is due solely to f having the smallest $c^\ell(\theta, f)$ or largest $\Delta E^\ell(f \mid \theta)$ value.

Unlike the global stability definition, the global instability treats each item in the following manner: Those f having smaller $\Delta E^\ell(f \mid \theta)$ (larger $c^\ell(\theta, f)$) values affect $C_p^\ell(\theta)$ in a more significant way. For $p = 2$, for example, the partial derivative is

$$\begin{aligned} \frac{\partial C_2^\ell(\theta)}{\partial \theta_k} &= \frac{\partial C_2^\ell(\theta)}{\partial \Delta E^\ell(f \mid \theta)} \frac{\partial \Delta E^\ell(f \mid \theta)}{\partial \theta_k} && (8.14) \\ &= \frac{1}{\left[\Delta E^\ell(f \mid \theta)\right]^3} [U_k(f) - U_k(\bar{f}^\ell)] && (8.15) \end{aligned}$$

where $E^\ell(f \mid \theta)$ takes the linear form (8.2). The smaller the $\Delta E^\ell(f \mid \theta)$ is, the more it affects θ. This is desirable because such f are more likely than the others to violate the correctness, because their $\Delta E^\ell(f \mid \theta)$ values are small, and should be more influential in determining θ.

[3] This is because, for the p-norm defined by $\|y\|_p = (|y_1|^p + |y_2|^p + \cdots + |y_n|^p)^{1/p}$, we have $\lim_{p\to\infty} \|y\|_p = \max_j |y_j|$.

Optimality

The optimal estimate is defined as the one in $\Theta_{correct}$ that minimizes the instability

$$\bar{\theta} = \arg \min_{\theta \in \Theta_{correct}} C_p(\theta) \tag{8.16}$$

Obviously, $C_p(\theta)$ is positive for all $\theta \in \Theta_{correct}$, and hence the minimal solution always exists. The minimal solution tends to increase $\Delta E(f \mid \theta)$ values in the global sense and thus maximizes the extent to which an example configuration $\bar{f}$ remains to be the global energy minimum when the observation d is perturbed. It is also expected that with such a $\bar{\theta}$, local minima corresponding to some low-energy-valued f are least likely to occur in minimization.

The correctness in (8.7), instability in (8.11), and optimality in (8.16) are defined without specifying the form of the energy $E(f \mid \theta)$. Therefore, the principle established so far is general for any optimization-based recognition models. Minimizing the instability with the constraint of the correctness is a nonlinear programming problem when the instability is nonlinear in θ.

For conciseness, in the following, the superscript ℓ will be omitted most of the time unless necessary.

8.2.3 Linear Classification Function

In this work, we are interested in cases where the $E(f \mid \theta)$ is linear in θ. Assume that, with a nonparametric or partially parametric modeling method, the energy is derived to take the linear form (8.2). With the linear form, the energy change can be written as

$$\Delta E(f \mid \theta) = \theta^T x(f) \tag{8.17}$$

where

$$x(f) = [x_0(f), x_1(f), \ldots, x_K(f)]^T = U(f) - U(\bar{f}) \tag{8.18}$$

is the potential change. Denote the set of all potential changes by

$$\mathcal{X} = \{x(f) \mid f \in \mathcal{F}\} \tag{8.19}$$

Note that $\mathcal{X}$ *excludes* $x(\bar{f})$, the vector of zeros. The set $\mathcal{X}$ will be used as the *training data set.* When there are $L > 1$ examples,

$$\mathcal{X} = \{x^\ell(f) = U^\ell(f) - U^\ell(\bar{f}^\ell) \mid f \in \mathcal{F}^\ell, \forall \ell\} \tag{8.20}$$

contains training data from all the instances. $\mathcal{X}$ is the $K+1$-dimensional data space in which x's are points.

The correctness (8.7) can be written as

$$\theta^T x(f) > 0 \qquad \forall x \in \mathcal{X} \tag{8.21}$$

In pattern recognition, $\theta^T x$ is called a *linear classification function* when considered as a function of x; $\theta^T x(f)$ is called a *generalized linear classification*

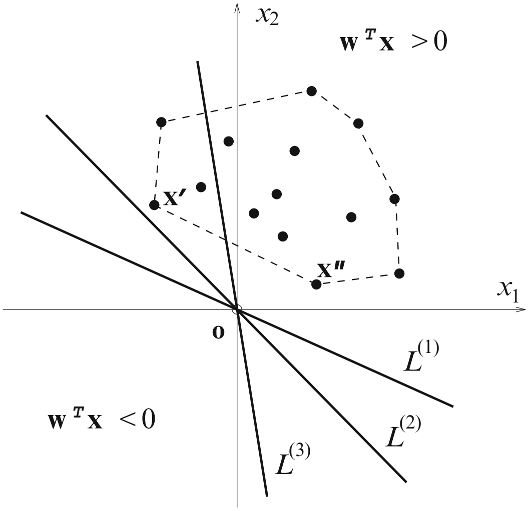

Figure 8.1: Correct and incorrect parameter estimates. A case where the C_∞-optimal hyperplane is parallel to $\overline{x'x''}$. From (Li 1997b) with permission; ©1997 Kluwer.

function when considered as a function of f. There exist useful theories and algorithms for linear pattern classification functions (Duda and Hart 1973).

The equation $\theta^T x = 0$ is a hyperplane in the space $\mathcal{X}$, passing through the origin $x(\bar{f}) = \mathbf{0}$. With a correct θ, the hyperplane divides the space into two parts, with all $x(f)$ ($f \in \mathcal{F}$) on one side, more exactly the "positive" side, of the hyperplane. The Euclidean distance from x to the hyperplane is equal to $\theta^T x/\|\theta\|$, a *signed* quantity. After the normalization (8.4), the point-to-hyperplane distance is just $\theta^T x(f)$.

The correctness can be judged by checking the minimal distance from the point set $\mathcal{X}$ to the hyperplane. We define the "separability" as the smallest distance value

$$S(\theta) = \min_f \theta^T x(f)/\|\theta\| \tag{8.22}$$

and it can also be considered as the stability of the system with a given θ. The correctness is equivalent to the positivity of the separability.

It is helpful to visually illustrate the optimality using C_∞. With C_∞, the minimal-instability solution is the same as the minimax solution

$$\bar{\theta} = \arg \min_{\theta \in \Theta_{correct}} C_\infty(\theta) = \arg \min_{\theta \in \Theta_{correct}} \left[\max_{f \in \mathcal{F}} \Delta E(f \mid \theta) \right] \tag{8.23}$$

In this case, $S(\theta) = 1/C_\infty(\theta)$ and the above is maximal separability.

Figure 8.1 qualitatively illustrates correct/incorrect and maximal separability parameters. It is an example in two-dimensional space where an energy

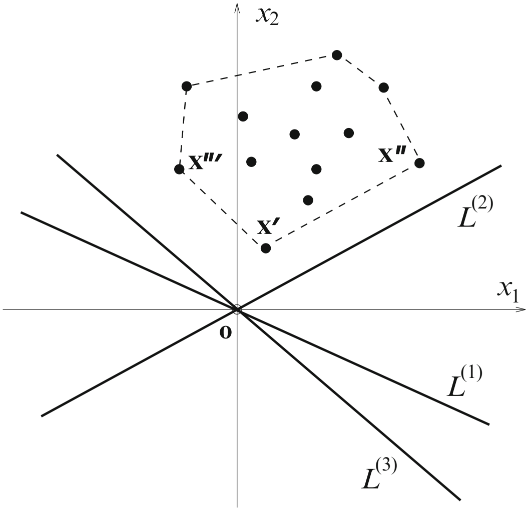

Figure 8.2: A case where the optimal hyperplane is perpendicular to $\overline{ox'}$. From (Li 1997b) with permission; ©1997 Kluwer.

change takes the form of $\Delta E(f \mid \theta) = \theta^T x(f) = \theta_1 x_1 + \theta_2 x_2$. The point $x(\bar{f}) = \mathbf{0}$ coincides with the origin of the x_1–x_2 space. Data $x(f)$ $(f \in \mathcal{F})$ are shown as filled dots. The three lines $L^{(1)}$, $L^{(2)}$, and $L^{(3)}$ represent three hyperplanes, corresponding to three different estimates of parameters θ. Parameter estimates for $L^{(1)}$ and $L^{(2)}$ are correct ones because they make all the data points on the positive side of the hyperplane and thus satisfy (8.7). However, $L^{(3)}$ is not a correct one. Of the two correct estimates in Fig. 8.1, the one corresponding to $L^{(1)}$ is better than the other in terms of C_∞.

The separability determines the range of disturbances in x within which the example configuration $\bar{f}$ remains minimal. Refer to point $x' = x(f')$ in the figure. Its distance to $L^{(2)}$ is the smallest. The point may easily deviate across $L^{(2)}$ to the negative half space due to some perturbation in the observation d. When this happens, $\Delta E(f \mid \theta) < 0$, causing $E(f' \mid \theta)$ to be lower than $E(\bar{f} \mid \theta)$. This means that $\bar{f}$ is no longer the energy minimum. If the parameters are chosen as those corresponding to $L^{(1)}$, the separability is larger and violation of the correctness is less likely to happen.

The dashed polygon in the figure forms the convex hull (polytope) of the data set. Only those data points that form the hull affect the minimax solution whereas those inside the hull are ineffective. The ineffective data points inside the hull can be removed and only the effective points, the number of which may be small compared with the whole data set, need be considered in the solution-finding process. This increases the efficiency.

There are two possibilities for the orientation of the maximal-separability hyperplane w.r.t. the polytope. The first is that the optimal hyperplane is parallel to one of the sides (faces in cases of 3D or higher-dimensional parameter space) of the polytope, which is the case in Fig. 8.1. The other is that the optimal hyperplane is perpendicular to the line linking the origin ø and the nearest data point, which is the case in Fig. 8.2. This is a property we may use to find the minimax solution. When the points constituting the polytope are identified, all the possible solutions can be enumerated; when there are only a small number of them, we can find the minimax solution by an exhaustive comparison. In Fig. 8.2, $L^{(2)}$ is parallel to $\overline{x'x''}$ and $L^{(3)}$ is parallel to $\overline{x'x'''}$, but $L^{(1)}$ is perpendicular to $\overline{\mathbf{0}x'}$. Let θ_1, θ_2 and θ_3 be the sets of parameters corresponding to $L^{(1)}$, $L^{(2)}$ and $L^{(3)}$. Suppose that in the figure the following relations hold: $S(\theta_1) > S(\theta_2)$ and $S(\theta_1) > S(\theta_3)$. Then θ_1 is the best estimate because it maximizes the separability.

The minimax solution (8.23) with $p = \infty$ was used in the above only for illustration. With $p = \infty$, a continuous change in x may lead to a discontinuous change in the C_∞-optimal solution. This is due to the decision-making nature of the definition which may cause discontinuous changes in the minimum. We use other p-instability definitions, with $p = 1$ or 2 for example, because they give more stable estimates.

8.2.4 A Nonparametric Learning Algorithm

Consider the case where $p = 2$. Because $\theta^T x \geq 0$, minimizing $C_2(\theta)$ is equivalent to minimizing its square,

$$[C_2(\theta)]^2 = \sum_{x \in \mathcal{X}} \frac{1}{(\theta^T x)^2} \tag{8.24}$$

The problem is given formally as

$$\begin{array}{ll} \text{minimize} & \sum_{x \in \mathcal{X}} \frac{1}{(\theta^T x)^2} \\ \text{subject to} & \|\theta\| = 1 \\ & \theta^T x > 0 \quad \forall x \in \mathcal{X} \end{array} \tag{8.25}$$

This is a nonlinear programming problem and can be solved using the standard techniques. In this work, a gradient-based, nonparametric algorithm is used to obtain a numerical solution.

The perceptron algorithm (Rosenblatt 1962) has already provided a solution for learning a correct parameter estimate. It iterates on θ to increase the objective of the form $\theta^T x$ based on the gradient information, where the gradient is simply x. In one cycle, all $x \in \mathcal{X}$ are tried in turn to adjust θ. If $\theta^T x > 0$, meaning x is correctly classified under θ, no action is taken. Otherwise, if $\theta^T x \leq 0$, θ is adjusted according to the gradient-ascent rule

$$\theta \longleftarrow \theta + \mu x \tag{8.26}$$

where μ is a small constant. The update is followed by a normalization operation

```
**Algorithm** learning($\theta$, $\mathcal{X}$)
/* Learning correct and optimal $\theta$ from the training data $\mathcal{X}$ */
Begin Algorithm
    initialize($\theta$);
    do {
        $\theta_{last} \leftarrow \theta$;
        if ($\theta^T x > 0 \;\; \exists x \in \mathcal{X}$) {
            correct($\theta$);
            $\theta \leftarrow \theta / \|\theta\|$;
        }
        $\theta \longleftarrow \theta + \mu \sum_{x \in \mathcal{X}} \frac{x}{(\theta^T x)^3}$;
        $\theta \leftarrow \theta / \|\theta\|$;
    } until ($\|\theta_{last}^T - \theta\| < \epsilon$);
    return($\bar{\theta} = \theta$);
End Algorithm
```

Figure 8.3: Algorithm for finding the optimal combination of parameters. From (Li 1997b) with permission; ©1997 Kluwer.

$$\theta \leftarrow \theta / \|\theta\| \tag{8.27}$$

when $\|\theta\| = 1$ is required. The cycle is repeated until a correct classification is made for all the data and thus a correct θ is learned. With the assumption that the solution exists, also meaning that the data set $\mathcal{X}$ is linearly separable from the origin of the data space, the algorithm is guaranteed to converge after a finite number of iterations (Duda and Hart 1973).

The objective function $[C_2(\theta)]^2 = \sum_{x \in \mathcal{X}} \frac{1}{(\theta^T x)^2}$ can be minimized in a similar way. The gradient is

$$\nabla [C_2(\theta)]^2 = -2 \sum_{x \in \mathcal{X}} \frac{x}{(\theta^T x)^3} \tag{8.28}$$

An update goes as

$$\theta \longleftarrow \theta + \mu \sum_{x \in \mathcal{X}} \frac{x}{(\theta^T x)^3} \tag{8.29}$$

where μ is a small constant. If θ were unconstrained, the above might diverge when $(\theta^T x)^3$ becomes too small. However, the update is again followed by the normalization $\|\theta\| = 1$. This process repeats until θ converges to $\bar{\theta}$.

In our implementation, the two stages are combined into one procedure as shown in Fig. 8.3. In the procedure, initialize(θ) sets θ at random, correct(θ)

learns a correct θ from $\mathcal{X}$, and $(\|\theta_{last}^T - \theta\| < \epsilon)$, where $\epsilon > 0$ is a small number, verifies the convergence. The algorithm is very stable.

The amount of change in each minimization iteration is $\mu \sum_{x \in \mathcal{X}} \frac{x}{(\theta^T x)^3}$. The influence from x is weighted by $1/(\theta^T x)^3$. This means that those x with smaller $\theta^T x$ values (closer to the hyperplane) have bigger force in pushing the hyperplane away from themselves, whereas those with big $\theta^T x$ values (far away from the hyperplane) have small influence. This effectively stabilizes the learning process.

When $p = \infty$, we are facing the minimax problem (8.23). An algorithm for solving this is the "generalized portrait technique" (Vapnik 1982), which is designed for constructing hyperplanes with maximum separability. It is extended by Boser, Guyon, and Vapnik (1992) to train classifiers of the form $\theta^T \varphi(x)$, where $\varphi(x)$ is a vector of functions of x. The key idea is to transform the problem into the dual space by means of the Lagrangian. This gives a quadratic optimization with constraints. The optimal parameter estimate is expressed as a linear combination of supporting patterns, where the supporting patterns correspond to the data points nearest to the hyperplane. Two benefits are gained from this method: There are no local minima in the quadratic optimization and the maximum separability obtained is insensitive to small changes of the learned parameters.

The θ computed using the nonparametric procedure is optimal w.r.t. the training data $\mathcal{X}$. It is the best result that can be obtained from $\mathcal{X}$ for generalization to other data. Better results may be obtained, provided that more knowledge about the training data is available.

8.2.5 Reducing Search Space

The data set $\mathcal{X}$ in (8.20) may be very large because there are a combinatorial number of possible configurations in $\mathcal{F} = \{f \neq \bar{f}\}$. In principle, all $f \neq \bar{f}$ should be considered. However, we assume that the configurations thay are neighboring $\bar{f}$ have the largest influence on the selection of θ. Define the neighborhood of $\bar{f}$ as

$$\mathcal{N}_{\bar{f}} = \{f = (f_1, \dots, f_m) \mid f_i \neq \bar{f}_i, f_i \in \mathcal{L}, \exists^1 i \in \mathcal{S}\} \tag{8.30}$$

where $\exists^1$ reads "one and only one exists" and $\mathcal{L}$ is the set of admissible labels for every f_i. The $\mathcal{N}_{\bar{f}}$ consists of all $f \in \mathcal{F}$ that differ from $\bar{f}$ by one and only one component. This confinement reduces the search space to an enormous extent. After the configuration space is confined to $\mathcal{N}_{\bar{f}}$, the set of training data is computed as

$$\mathcal{X} = \{x = U(f) - U(\bar{f}) \mid f \in \mathcal{N}_{\bar{f}}\} \tag{8.31}$$

which is much smaller than the $\mathcal{X}$ in (8.20).

8.3 Application in MRF Object Recognition

The theory is applied to the parameter estimation for MRF object recognition where the form of the energy is derived based on MRF's. MRF modeling provides one approach to optimization-based object recognition (Modestino and Zhang 1989; Cooper 1990; Baddeley and van Lieshout 1992; Kim and Yang 1992; Li 1994a). The MAP solution is usually sought. The posterior distribution of configurations f is of Gibbs type

$$P(f \mid d) = Z^{-1} \mathrm{e}^{-E(f \mid \theta)} \tag{8.32}$$

where Z is a normalizing constant called the partition function and $E(f \mid \theta)$ is the posterior energy function measuring the global cost of f. In the following, $E(f \mid \theta)$ is defined and converted to the linear form $\theta^T U(f)$.

8.3.1 Posterior Energy

An object or a scene is represented by a set of features where the features are attributed by their properties and constrained to one another by contextual relations. Let a set of m features (sites) in the scene be indexed by $\mathcal{S} = \{1, \ldots, m\}$, a set of M features (labels) in the considered model object by $\mathcal{L} = \{1, \ldots, M\}$, and everything in the scene not modeled by labels in $\mathcal{L}$ by $\{0\}$ which is a virtual NULL label. The set union $\mathcal{L}^+ = \mathcal{L} \cup \{0\}$ is the augmented label set. The structure of the scene is denoted by $\mathcal{G} = (\mathcal{S}, d)$ and that of the model object by $\mathcal{G}' = (\mathcal{L}, D)$, where d denotes the visual constraints on features in $\mathcal{S}$ and D describes the visual constraints on features in $\mathcal{L}$, where the constraints can be, for example, properties and relations between features.

Let object recognition be posed as assigning a label from $\mathcal{L}^+$ to each of the sites in $\mathcal{S}$ so as to satisfy the constraints. The labeling (configuration) of the sites is defined by $f = \{f_1, \ldots, f_m\}$, in which $f_i \in \mathcal{L}^+$ is the label assigned to i. A pair $(i \in \mathcal{S}, f_i \in \mathcal{L}^+)$ is a match or correspondence. Under contextual constraints, a configuration f can be interpreted as a mapping from the structure of the scene $\mathcal{G} = (\mathcal{S}, d)$ to the structure of the model object $\mathcal{G}' = (\mathcal{L}, D)$. Therefore, such a mapping is denoted as a triple $(f, \mathcal{G}, \mathcal{G}')$.

The observation $d = (d_1, d_2)$, which is the feature extracted from the image, consists of two sources of constraints, unary properties d_1 for single-site features, such as color and size, and binary relations d_2 for pair-site features, such as angle and distance. More specifically, each site $i \in \mathcal{S}$ is associated with a set of K_1 properties $\{d_1^{(k)}(i) \mid k = 1, \ldots, K_1, i \in \mathcal{S}\}$ and each pair of sites with a set of K_2 relations $\{d_2^{(k)}(i, i') \mid k = 1, \ldots, K_2; i, i' \in \mathcal{S}\}$. In the model object library, we have model features $\{D_1^{(k)}(I) \mid k = 1, \ldots, K_1, I \in \mathcal{L}\}$ and $\{D_2^{(k)}(I, I') \mid k = 1, \ldots, K_2; I, I' \in \mathcal{L}\}$ (note that $\mathcal{L}$ excludes the NULL label). According to (4.14), under the labeling f, the observation d is a noise-contaminated version of the corresponding model features D

$$d_1(i) = D_1(f_i) + e(i), \qquad d_1(i,i') = D_2(f_i, f_{i'}) + e(i,i') \tag{8.33}$$

where $f_i, f_{i'} \neq 0$ are nonNULL matches and e is a white Gaussian noise; that is, $d_1(i)$ and $d_2(i,i')$ are white Gaussian distributions with conditional means $D_1(f_i)$ and $D_2(f_i, f_{i'})$, respectively.

The posterior energy $E(f) = U(f \mid d)$ takes the form shown in (4.18), rewritten as

$$\begin{aligned} E(f) = \; & \textstyle\sum_{i\in\mathcal{S}} V_1(f_i) + \\ & \textstyle\sum_{i\in\mathcal{S}} \sum_{i'\in\mathcal{N}_i} V_2(f_i, f_{i'}) + \\ & \textstyle\sum_{i\in\mathcal{S}: f_i\neq 0} V_1(d_1(i) \mid f_i) + \\ & \textstyle\sum_{i\in\mathcal{S}: f_i\neq 0} \sum_{i'\in\mathcal{S}-\{i\}: f_{i'}\neq 0} V_2(d_2(i,i') \mid f_i, f_{i'}) \end{aligned} \tag{8.34}$$

The first and second summations are due to the joint prior probability of the MRF labels f; the third and fourth are due to the conditional p.d.f. of d or the likelihood of f, respectively. Refer to (4.11), (4.12), (4.16), and (4.17).

8.3.2 Energy in Linear Form

The parameters involved in $E(f)$ are the noise variances $[\sigma_n^{(k)}]^2$ and the prior penalties v_{n0} ($n = 1, 2$). Let the parameters be denoted uniformly by $\theta = \{\theta_n^{(k)} \mid k = 0, \dots, K_n, n = 1, 2\}$. For $k = 0$,

$$\theta_n^{(0)} = v_{n0} \tag{8.35}$$

and for $k \geq 1$,

$$\theta_n^{(k)} = (2[\sigma_n^{(k)}]^2)^{-1} \tag{8.36}$$

Note all $\theta_n^{(k)} \geq 0$. Let the different energy components be uniformly denoted by $U = \{U_n^{(k)} \mid k = 0, \dots, K_n, n = 1, 2\}$. For $k = 0$,

$$U_1^{(0)}(f) = N_1 = \#\{f_i = 0 \mid i \in \mathcal{S}\} \tag{8.37}$$

is the number of NULL labels in f and

$$U_2^{(0)}(f) = N_2 = \#\{f_i = 0 \text{ or } f_{i'} = 0 \mid i \in \mathcal{S}, i' \in \mathcal{N}_i\} \tag{8.38}$$

is the number of label pairs, at least one of which is NULL . For $k \geq 1$, $U_n^{(k)}(f)$ relates to the likelihood energy components; They measure how much the observations $d_n^{(k)}$ deviate from the values $D_n^{(k)}$ that should-be true under f:

$$U_1^{(k)}(f) \stackrel{\triangle}{=} U_1^{(k)}(d \mid f) = \sum_{i\in\mathcal{S}, f_i\neq 0} [d_1^{(k)}(i) - D_1^{(k)}(f_i)]^2 / \{2[\sigma_1^{(k)}]^2\} \tag{8.39}$$

and

$$\begin{aligned} U_2^{(k)}(f) &\stackrel{\triangle}{=} U_2^{(k)}(d \mid f) \\ &= \sum_{i \in \mathcal{S}, f_i \neq 0} \sum_{i' \in \mathcal{S}, i' \neq i, f_{i'} \neq 0} [d_2^{(k)}(i,i') - D_2^{(k)}(f_i, f_{i'})]^2 / \{2[\sigma_2^{(k)}]^2\} \end{aligned} \tag{8.40}$$

After some manipulation, the energy can be written as

$$E(f \mid \theta) = \sum_{n=1}^{2} \sum_{k=0}^{K_n} \theta_n^{(k)} U_n^{(k)}(f) = \theta^T U(f) \tag{8.41}$$

where θ and $U(f)$ are column vectors of $K_1 + K_2 + 2$ components. Given an instance $(\bar{f}, \mathcal{G}, \mathcal{G}')$, the $U(\bar{f})$ is a *known* vector of real numbers. The θ is the vector of *unknown* weights to be determined. The stability follows immediately as

$$\Delta E(f \mid \theta) = \theta^T x(f) \tag{8.42}$$

where $x(f) = U(f) - U(\bar{f})$.

Some remarks on θ, $U(f)$, and E are in order. Obviously, all $\theta_n^{(k)}$ and $U_n^{(k)}$ are nonnegative. In the ideal case of exact (possibly partial) matching, all $U_n^{(k)}(f)$ $(k \geq 1)$ are zeros because $d_1^{(k)}(i)$ and $D_1^{(k)}(f_i)$ are exactly the same and so are $d_2^{(k)}(i,i')$ and $D_2^{(k)}(f_i, f_{i'})$. In the general case of inexact matching, the sum of the $U_n^{(k)}$ should be as small as possible for the minimal solution. The following are some properties of E:

- Given f, $E(f \mid \theta)$ is linear in θ. Given θ, it is linear in $U(f)$.
- For $\kappa > 0$, θ and $\kappa\theta$ are equivalent, as has been discussed.
- The values of $\theta_n^{(0)}$ relative to those of $\theta_n^{(k)}$ $(k \geq 1)$ affect the rate of NULL labels. The higher the penalties $\theta_n^{(0)}$ are, the more sites in $\mathcal{S}$ will be assigned nonNULL labels and vice versa.

The first property enables us to use the results we established for linear classifiers in learning the correct and optimal θ. According to the second property, a larger $\theta_n^{(k)}$ relative to the rest makes the constraints $d_n^{(k)}$ and $D_n^{(k)}$ play a more important role. Useless and misleading constraints $d_n^{(k)}$ and $D_n^{(k)}$ should be weighted by 0. Using the third property, one can decrease $\theta_n^{(0)}$ values to increase the number of the NULL labels. This is because for the minimum energy matching, a lower cost for NULL labels makes more sites labeled NULL , which is equivalent to discarding more not so reliable nonNULL labels into the NULL bin.

8.3.3 How the Minimal Configuration Changes

The following analysis examines how the minimum $f^* = \arg\min_f E(f \mid \theta)$ changes as the observation changes from $d0$ to $d = d0 + \delta d$, where δd is a perturbation. In the beginning, when $\|\delta d\|$ is close to 0, f^* should remain as the minimum for a range of such small δd. This is simply because $E(f \mid \theta)$ is continuous w.r.t. d. When the perturbation becomes larger and larger, the minimum has to give way to another configuration.

When should a change happen? To see the effect more clearly, assume the perturbation is in observation components related to only a particular i so that the only changes are $d0_1^{(k)}(i) \rightarrow d_1^{(k)}(i)$ and $d0_2^{(k)}(i,i') \rightarrow d_2^{(k)}(i,i')$, $\forall i' \in \mathcal{N}_i$. First, assume that f_i^* is a nonNULL label ($f_i^* \neq 0$) and consider such a perturbation δd that incurs a *larger* likelihood potential. Obviously, as the likelihood potential (conditioned on $\{f_{i'}^* \neq 0 \mid i' \in \mathcal{N}_i\}$)

$$\begin{aligned} V(d \mid f_i^*) \quad = \quad & \sum_{k=1}^{K_1} \theta_1^{(k)} (d_1^{(k)}(i) - D_1^{(k)}(f_i^*))^2 + \\ & \sum_{i' \in \mathcal{S}, i' \neq i, f_{i'}^* \neq 0} \sum_{k=1}^{K_2} \theta_2^{(k)} (d_2^{(k)}(i,i') - D_2^{(k)}(f_i^*, f_{i'}^*))^2 \end{aligned} \tag{8.43}$$

increases, it will eventually become cheaper for $f_i^* \neq 0$ to change to $f_i = 0$. More accurately, this should happen when

$$V(d \mid f_i^*) > \theta_1^{(0)} + \sum_{i' \in \mathcal{N}_i, f_{i'}^* \neq 0} \theta_2^{(0)} = \theta_1^{(0)} + N_2^i \theta_2^{(0)} \tag{8.44}$$

where

$$N_2^i = \#\{i' \in \mathcal{N}_i \mid f_{i'}^* \neq 0\} \tag{8.45}$$

is the number of nonNULL labeled sites in $\mathcal{N}_i$ under f^*.

Next, assume $f_i^* = 0$ and consider such a perturbation that incurs a *smaller* likelihood potential. The perturbation has to be such that d and D more closely resemble each other. As the conditional likelihood potential

$$\begin{aligned} V(d \mid f_i) = \quad & \textstyle\sum_{k=1}^{K_1} \theta_1^{(k)} (d_1^{(k)}(i) - D_1^{(k)}(f_i))^2 + \\ & \textstyle\sum_{i' \in \mathcal{S}, i' \neq i, f_{i'}^* \neq 0} \sum_{k=1}^{K_2} \theta_2^{(k)} (d_2^{(k)}(i,i') - D_2^{(k)}(f_i, f_{i'}^*))^2 \end{aligned} \tag{8.46}$$

decreases, it will eventually become cheaper for $f_i^* = 0$ to change to one of the nonNULL labels, $f_i \neq 0$. More accurately, this should happen when

$$V(d \mid f_i) < \theta_1^{(0)} + \sum_{i' \in \mathcal{N}_i, f_{i'}^* \neq 0} \theta_2^{(0)} = \theta_1^{(0)} + N_2^i \theta_2^{(0)} \tag{8.47}$$

The analysis above shows how the minimal configuration f^* adjusts as d changes when θ is fixed. On the other hand, the f^* can be maintained unchanged by adjusting θ; this means a different encoding of constraints into E.

8.3.4 Parametric Estimation under Gaussian Noise

Assuming the functional form of the noise distribution is known, then we can take advantage of (partial) parametric modeling for the estimation. When the noise is additive white Gaussian with unknown $[\sigma_n^{(k)}]^2$, the estimate can be obtained in closed form. The closed form estimation is performed in two steps. (1) Estimate the noise variances $[\bar{\sigma}_n^{(k)}]^2$ $(k \geq 1)$ and then compute the weights $\bar{\theta}_n^{(k)}$ $(k \geq 1)$ using the relationship in (8.36). (2) Then compute the allowable $\bar{\theta}_n^{(0)}$ relative to $\bar{\theta}_n^{(k)}$ $(k \geq 1)$ to satisfy the correctness in (8.7). Optimization like (8.16) derived using a nonparametric principle may not be applicable in this case.

Given d, D, and $\bar{f}$, the Gaussian noise variances can be estimated by maximizing the joint likelihood function $p(d \mid \bar{f})$ (ML estimation). The ML estimates are simply

$$[\bar{\sigma}_1^{(k)}]^2 = \frac{1}{N_1'} \sum_{i \in \mathcal{S}, \bar{f}_i \neq 0} [d_1^{(k)}(i) - D_1^{(k)}(\bar{f}_i)]^2 \tag{8.48}$$

and

$$[\bar{\sigma}_2^{(k)}]^2 = \frac{1}{N_2'} \sum_{i \in \mathcal{S}, \bar{f}_i \neq 0} \sum_{i' \in \mathcal{S}, i' \neq i, \bar{f}_{i'} \neq 0} [d_2^{(k)}(i,i') - D_2^{(k)}(\bar{f}_i, \bar{f}_i)]^2 \tag{8.49}$$

where

$$N_1' = \#\{i \in \mathcal{S} \mid f_i \neq 0\} \tag{8.50}$$

is the number of nonNULL labels in f and

$$N_2' = \#\{(i,i') \in \mathcal{S}^2 \mid i' \neq i,\ f_i \neq 0,\ f_{i'} \neq 0\} \tag{8.51}$$

is the number of label pairs of which neither is the NULL . The optimal weights for $k \geq 1$ can be obtained immediately by

$$\bar{\theta}_n^{(k)} = 1/2[\bar{\sigma}_n^{(k)}]^2 \tag{8.52}$$

So far, only the example configurations $\bar{f}$, not others, are used in computing the $\bar{\theta}_n^{(k)}$.

Now the remaining problem is to determine $\bar{\theta}_n^{(0)}$ to meet the correctness (8.7). Because $\theta_n^{(0)} = v_{n0}$, this is done to estimate the MRF parameters v_{n0} in the prior distributions implied in the given examples. There may be a *range* of $\bar{\theta}_n^{(0)}$ under which each $\bar{f}$ is correctly encoded. The range is determined by the lower and upper bounds.

In doing so, only those configurations in $\mathcal{N}_{\bar{f}}$ that reflect transitions from a nonNULL to the NULL label and the other way around are needed; the other configurations, which reflect transitions from one nonNULL label to another, are not. This subset is obtained by changing each of the nonNULL labels in $\bar{f}$ to the NULL label or changing each of the NULL labels to a nonNULL label.

First, consider label changes from a nonNULL label to the NULL label. Assume a configuration change from $\bar{f}$ to f is due to the change from $\bar{f}_i \neq 0$ to $f_i = 0$ for just one $i \in \mathcal{S}$. The corresponding energy change is given by

$$\begin{aligned}\tfrac{1}{2}\Delta E(f \mid \theta) = \; & \theta_1^{(0)} - \textstyle\sum_{k=1}^{K_1} \bar{\theta}_1^{(k)} [d_1^{(k)}(i) - D_1^{(k)}(\bar{f}_i)]^2 + \\ & \textstyle\sum_{i' \in \mathcal{N}_i, \bar{f}_{i'} \neq 0} \theta_2^{(0)} - \\ & \textstyle\sum_{i' \in \mathcal{S}, i' \neq i, \bar{f}_{i'} \neq 0} \sum_{k=1}^{K_2} \bar{\theta}_2^{(k)} [d_2^{(k)}(i,i') - D_2^{(k)}(\bar{f}_i, \bar{f}_{i'})]^2\end{aligned} \tag{8.53}$$

The change above must be positive, $\Delta E(f \mid \theta) > 0$. Suppose there are N nonNULL labeled sites under $\bar{f}$ and therefore $\bar{f}$ has N such neighboring configurations. Then N such inequalities of $\Delta E(f \mid \theta) > 0$ can be obtained. The two unknowns, $\theta_1^{(0)}$ and $\theta_2^{(0)}$, can be solved for and used as the lower bounds $(\theta_1^{(0)})_{min}$ and $(\theta_2^{(0)})_{min}$.

Similarly, the upper bounds can be computed by considering label changes from the NULL label to a nonNULL label. The corresponding energy change due to a change from $\bar{f}_i = 0$ to $f_i \neq 0$ is given by

$$\begin{aligned}\tfrac{1}{2}\Delta E(f \mid \theta) = \; & \textstyle\sum_{k=1}^{K_1} \bar{\theta}_1^{(k)} [d_1^{(k)}(i) - D_1^{(k)}(f_i)]^2 \quad - \quad \theta_1^{(0)} \\ & \textstyle\sum_{i' \in \mathcal{S}, \bar{f}_{i'} \neq 0} \sum_{k=1}^{K_2} \bar{\theta}_2^{(k)} [d_2^{(k)}(i,i') - D_2^{(k)}(f_i, f_{i'})]^2 - \\ & \textstyle\sum_{i' \in \mathcal{N}_i, \bar{f}_{i'} \neq 0} \theta_2^{(0)}\end{aligned} \tag{8.54}$$

The change above must also be positive, $\Delta E(f \mid \theta) > 0$. Suppose there are N NULL labeled sites under $\bar{f}$ and recall that there are M possible nonNULL labels in $\mathcal{L}$. Then $N \times M$ inequalities can be obtained. The two unknowns, $\theta_1^{(0)}$ and $\theta_2^{(0)}$, can be solved and used as the upper bounds $(\theta_1^{(0)})_{max}$ and $(\theta_2^{(0)})max$. If the example configurations $\bar{f}$ are minimal for the corresponding $\mathcal{G}$ and $\mathcal{G}'$, then the solution must be consistent; that is, $(\theta_n^{(0)})_{min} < (\theta_n^{(0)})_{max}$, for each instance.

Now, the space of all correct parameters is given by

$$\Theta_{correct} = \{\theta_n^{(k)} \mid \theta_n^{(0)} \in [(\theta_n^{(0)})_{min}, (\theta_n^{(0)})_{max}];\ \theta_n^{(k)} = \bar{\theta}_n^{(k)}, k \geq 1\} \tag{8.55}$$

A correct θ makes $\theta^T x > 0$ for all $x \in \mathcal{X}$. The hyperplane $\theta^T x = 0$ partitions $\mathcal{X}$ into two parts, with all $x \in \mathcal{X}$ on the positive side of it. The value for $\bar{\theta}_n^{(0)}$ may simply be set to the average $[(\theta_n^{(0)})_{min} + (\theta_n^{(0)})_{min}]/2$.

When there are $L > 1$ instances, $\theta_n^{(k)}$ $(k \geq 1)$ are obtained from the data set computed from all the instances. Given the common $\theta_n^{(k)}$, L correct ranges can be computed. The correct range for the L instances as a whole is the intersection of the L ranges. As a result, the overall $(\theta_n^{(0)})_{min}$ is the maximum of all the lower bounds and $(\theta_n^{(0)})_{max}$ the minimum of all the upper bounds. Although each range can often be consistent (i.e., $(\theta_n^{(0)})_{min} < (\theta_n^{(0)})_{max}$ for each n), there is less of a chance to guarantee that they, as a whole, are consistent for all $\ell = 1, \ldots, L$: The intersection may be empty when $L > 1$.

This inconsistency means a correct estimate does not exist for all the instances as a whole. There are several reasons for this. First of all, the assumptions, such as the model being Gaussian, are not verified by the data set, especially when the data set is small. In this case, the noise in different instances has different variances; when the ranges are computed under the assumption that the ML estimate is common to all instances, they may not be consistent with each other. This is the most direct reason for the inconsistency. Second, $\bar{f}$ in some examples cannot be embedded as the minimal energy configuration to satisfy the given constraints. Such instances are misleading and also cause inconsistency.

8.4 Experiments

The following experiments demonstrate: (i) the computation (learning) of the optimal parameter $\bar{\theta}$ from the examples given in the form of a triplet $(\bar{f}, \mathcal{G}, \mathcal{G}')$, and (ii) the use of the learned estimate $\bar{\theta}$ to recognize other scenes and models. The nonparametric learning algorithm is used because the data size is too small to assume a significant distribution. The convergence of the learning algorithm is demonstrated.

8.4.1 Recognition of Line Patterns

This experiment performs the recognition of simulated objects of line patterns under 2D rotation and translation. There are six possible model objects shown in Fig. 8.4. Figure 8.5 gives an example used for parameter estimation. The scene is given in the dotted and dashed lines, which are generated as follows. (1) Take a subset of lines from each of the three objects in Fig. 8.4(a)–(c); (2) rotate and translate each of the subsets; (3) mix the transformed subsets; (4) randomly deviate the positions of the endpoints of the lines, which results in the dotted lines; and (5) add spurious lines, shown as the dashed lines. The scene generated consists of several subsets of model patterns plus spurious lines, as shown in Fig. 8.5(b). The example configuration $\bar{f}$ is shown in Fig. 8.5(a). It maps the scene to one of the models given in Fig. 8.4. The alignment between the dotted lines of the scene and the solid lines of the model gives the nonNULL labels of $\bar{f}$, whereas the unaligned lines of the scene are labeled as NULL .

The following four types of bilateral relations are used with $n = 2$ and $K_2 = 4$):

(1) $d_2^{(1)}(i, i')$: the angle between lines i and i';

(2) $d_2^{(2)}(i, i')$: the distance between the mid-points of the lines;

(3) $d_2^{(3)}(i, i')$: the minimum distance between the endpoints of the lines; and

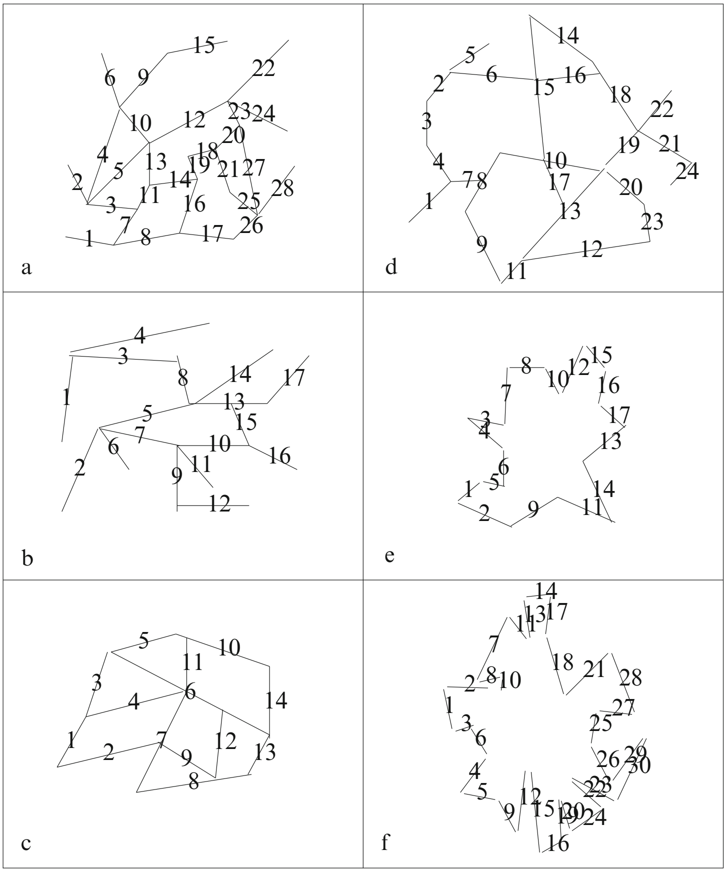

Figure 8.4: The six objects of line patterns in the model base. From (Li 1997b) with permission; ©1997 Kluwer.

(4) $d_2^{(4)}(i,i')$: the maximum distance between the endpoints of the lines.

Similarly, there are four model relations $D_2^{(k)}(I,I')$ ($k=1,\dots,4$) of the same type. No unary properties are used ($K_1=0$). The $\mathcal{G}$ and $\mathcal{G}'$ are composed of these four relational measurements. Therefore, there are five components ($k=0,1,\dots,4$) in x and θ.

The C_2-optimal parameters are computed as $\bar{\theta}=\{\bar{\theta}_2^{(0)},\bar{\theta}_2^{(1)},\dots,\bar{\theta}_2^{(4)}\}=\{0.58692,\ 0.30538,\ 0.17532,\ 0.37189,\ 0.62708\}$, which satisfies $\|\theta\|=1$. The computation takes a few seconds on an HP series 9000/755 workstation. To be used for recognition, $\bar{\theta}$ is multiplied by a factor of $0.7/\bar{\theta}_2^{(0)}$, yielding the final

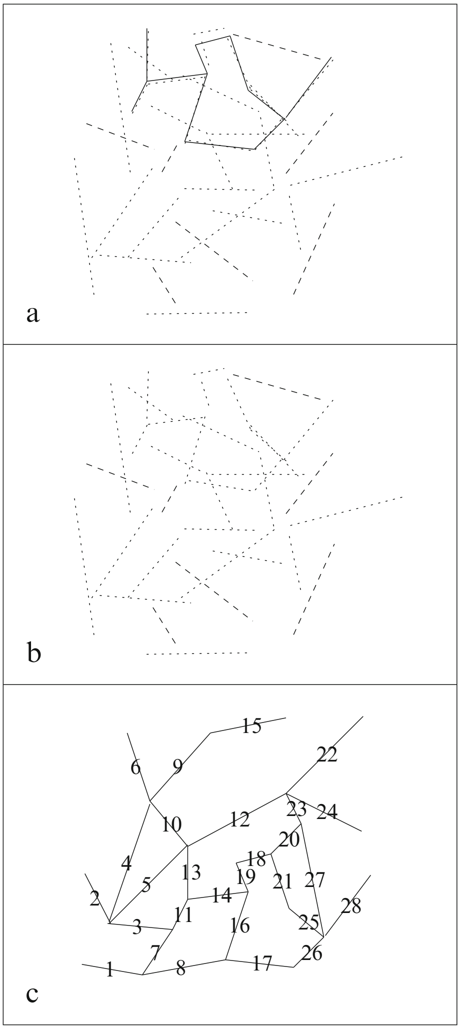

Figure 8.5: An exemplary instance consisting of (a) exemplary configuration $\bar{f}$, (b) scene $\mathcal{G}$, and (c) model $\mathcal{G}'$. From (Li 1997b) with permission; ©1997 Kluwer.

weights $\theta^* = \{0.70000, 0.36422, 0.20910, 0.44354, 0.74789\}$ (our recognition system requires $(\theta_2^{(0)})^* = 0.7$).

The θ^* is used to define the energy for recognizing other objects and scenes. The recognition results are shown in Fig. 8.6. There are two scenes, one in the upper row and the other in the lower row, composed of the dotted

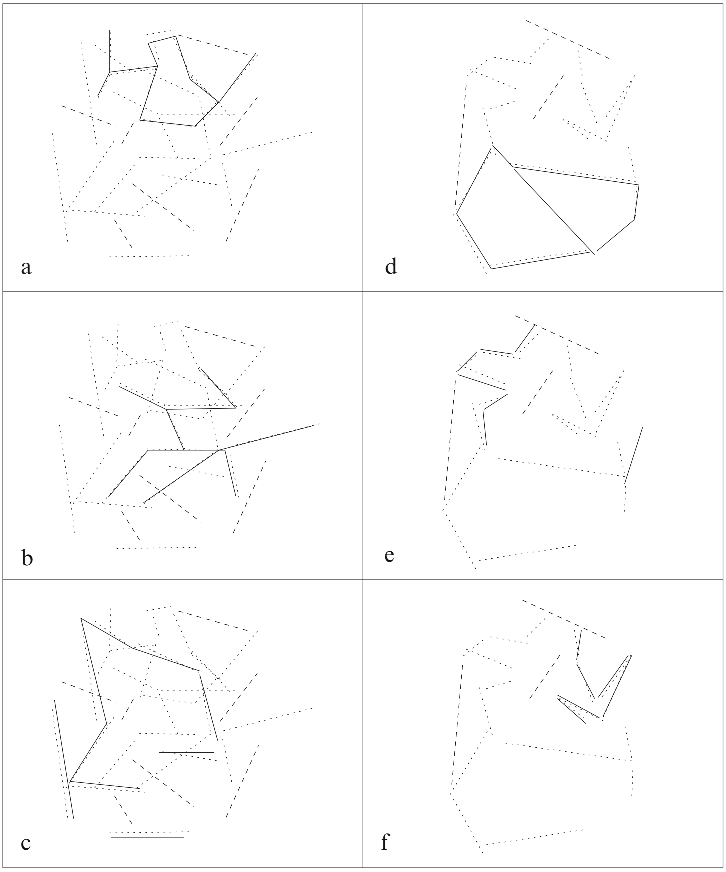

Figure 8.6: The optimal parameter estimate learned from the example is used to recognize other scenes and models (see the text). From (Li 1997b) with permission; ©1997 Kluwer.

and dashed lines. The upper one was used in the example, whereas the lower scene contains subparts of the three model objects in Fig. 8.4(d)–(f). Each scene is matched against the six model objects. The optimally matched object lines are shown as solid lines aligned with the scenes. The objects in the scenes are correctly matched to the model objects.

8.4.2 Recognition of Curved Objects

This experiment deals with jigsaw objects under 2D rotation, translation and uniform scaling. There are eight model jigsaw objects shown in Fig. 4.10. In this case of curved objects, the features of an object correspond to the corner points of its boundary. For both the scene and the models, the boundaries are extracted from the images using the Canny detector followed by hysteresis and edge linking. Corners are detected after that. No unary relations are used ($K_1 = 0$). Denoting the corners by $p_1, \ldots, p_m$, the following five types of bilateral relations are used ($n = 2; K_2 = 5$) based on a similarity-invariant curve representation of curves (Li 1993):

(1) $d_2^{(1)}(i, i')$: ratio of curve arc length $\widehat{p_i p_{i'}}$ and chord length $\overline{p_i p_{i'}}$;

(2) $d_2^{(2)}(i, i')$: ratio of curvature at p_i and $p_{i'}$;

(3) $d_2^{(3)}(i, i')$: invariant coordinate vector;

(4) $d_2^{(4)}(i, i')$: invariant radius vector; and

(5) $d_2^{(5)}(i, i')$: invariant angle vector.

They are computed using information about both the boundaries and the corners. Similarly, there are five model relations $D_2^{(k)}(I, I')$ ($k = 1, \ldots, 5$) of the same types. Therefore, there are six components ($k = 1, \ldots, 5$) in each x and θ, one for the NULL and five for the relational quantities above.

Figure 8.7 gives the example used for parameter estimation. The scene in Fig. 8.7(b) contains rotated, translated and scaled parts of one of the model jigsaw objects. Some objects in the scene are considerably occluded. The alignment between the model jigsaw object (the highlighted curve in (a)) and the scene gives nonNULL labels of $\bar{f}$, whereas the unaligned boundary corners of the scene are labeled as NULL .

The C_2-optimal parameters are computed as $\bar{\theta} = \{\bar{\theta}_2^{(0)}, \bar{\theta}_2^{(1)}, \ldots, \bar{\theta}_2^{(6)}\} = \{0.95540, 0.00034, 0.00000, 0.06045, 0.03057, 0.28743\}$, which satisfies $\|\theta\| = 1$. It takes a few seconds on the HP workstation. Note that the weight $\bar{\theta}_2^{(2)}$ for $d_2^{(2)}(i, i')$ and $D_2^{(2)}(I, I')$ (ratio of curvature) are zero. This means that this type of feature is not reliable enough to be used. Because our recognition system has a fixed value of $\theta_2^{(0)} = 0.7$, $\bar{\theta}$ is multiplied by a factor of $0.7/\bar{\theta}_2^{(0)}$, yielding the final weights $\theta^* = \{0.70000, 0.00025, 0.00000, 0.04429, 0.02240, 0.21060\}$.

The θ^* is used to define the energy function for recognizing other objects and scenes. The recognition results are shown in Fig. 8.8. There are two scenes, one in the left column and the other in the right column. The scene on the left was the one used in the example and the one on the right is a new scene. The optimally matched model objects are shown in the highlighted curves aligned with the scenes. The same results can also be obtained using the C_∞-optimal estimate, which is $\{0.7, 0.00366, 0.00000, 0.09466, 0.00251\}$.

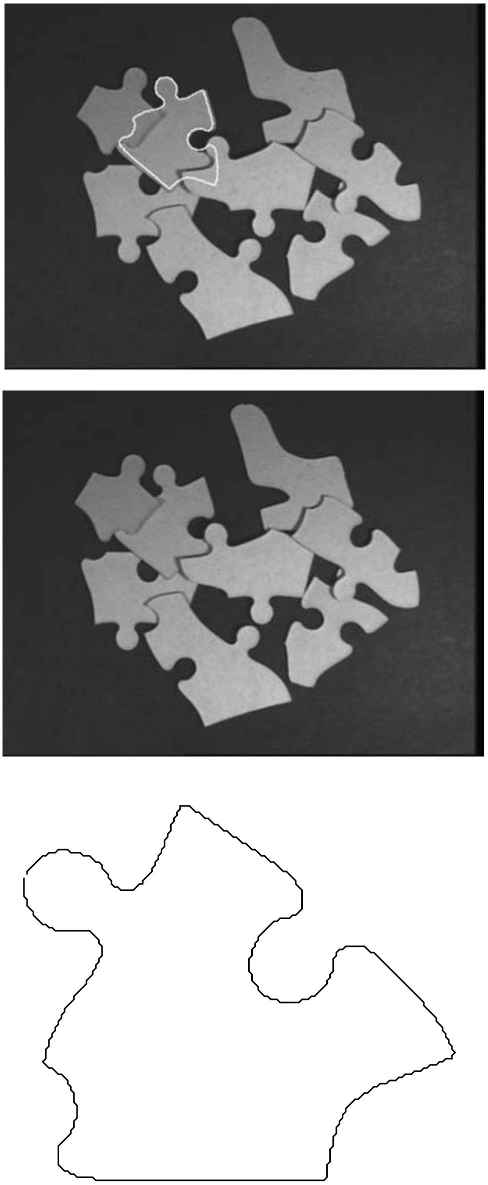

Figure 8.7: An example consisting of (a) example configuration $\bar{f}$, (b) scene $\mathcal{G}$ and (c) model $\mathcal{G}'$. From (Li 1997b) with permission; ©1997 Kluwer.

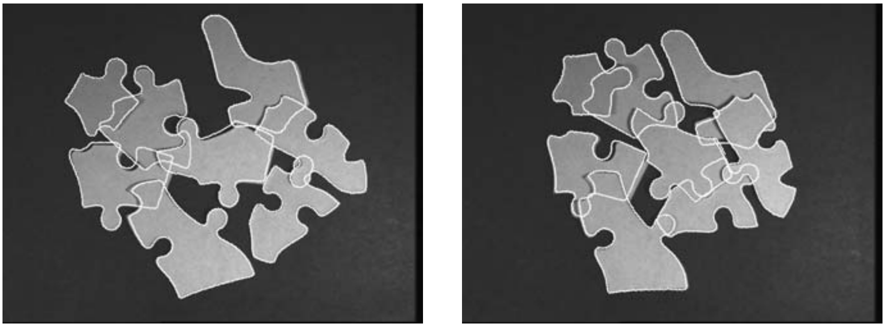

Figure 8.8: The learned estimate is used to recognize other scenes and models. The matched model jigsaw objects are aligned with the scene. From (Li 1997b) with permission; ©1997 Kluwer.

8.4.3 Convergence

The parameter estimation algorithm is very stable and has a nice convergence property. Figure 8.9 shows how the global instability measure C_2 and one of the learned parameters $\bar{\theta}_2^{(3)}$ evolve given different starting points. The values stabilize after hundreds of iterations, and different starting points converge to the same point.

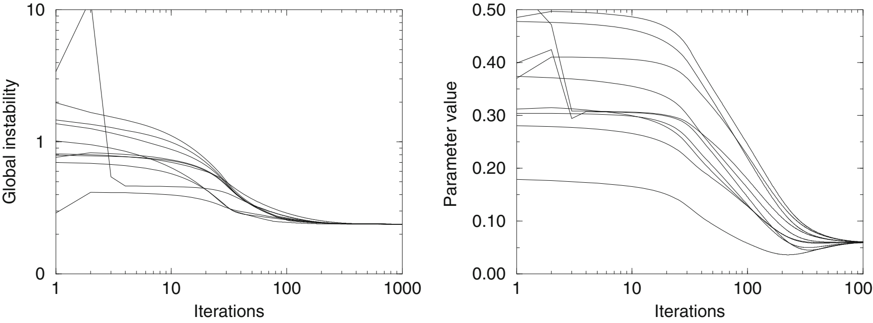

Figure 8.9: Convergence of the algorithm from different starting points. Left: Trajectories of C_2. Right: Trajectories of $\bar{\theta}_2^{(3)}$. From (Li 1997b) with permission; ©1997 Kluwer.

8.5 Conclusion

While manual selection is a common practice in object recognition systems, this chapter has presented a novel theory for automated optimal parameter estimation in optimization-based object recognition. The theory is based on learning from examples. Mathematical principles of correctness and instability are established and defined for the evaluation of parameter estimates. A learning algorithm is presented for computing the optimal (i.e., minimal-instability) estimate. An application to MRF-based recognition is given. Experiments conducted show very promising results. Optimal estimates automatically learned from examples can be well generalized for recognizing other scenes and objects.

The training examples are given to reflect the designer's judgment of desirable solutions. However, a recognizer with a given functional form cannot be trained by arbitrary examples. The example should be selected properly to reflect the correct semantics, in other words, they should be consistent with the constraints with which the functional form is derived. Assuming the form of the objective function is correct and the training set contains useful information, then the more examples are used for training, the more generalizable the learned parameter estimate will be.

The learning procedure also provides a means for checking the validity of the energy function derived from mathematical models. An improper mathematical model leads to an improper functional form. If no correct parameter estimates can be learned, it is a diagnostic symptom that the assumptions used in the model are not suitable for modeling the reality of the scene. The procedure also provides useful information for feature selection. Components of the optimal parameter estimate will be zero or near zero for unstable features.