HIERARCHICAL DESIGNS FOR PATTERN RECOGNITION

DONALD GEMAN

Dept. of Applied Mathematics and Statistics, and

Center for Imaging Science, Johns Hopkins University

- **Proposal:** *Make computational efficiency the organizing principle for machine perception.*
- **Motivation:**
  - Everyday experience (e.g., playing "20 Questions")
  - Small-sample learning
  - Machine vision experiments: With Y.Amit, F. Fleuret, X. Fan, F. Jung
- **Theoretical Analysis:** With G. Blanchard

MYSTERIES OF VISION

Not much is "obvious" in either computer or biological perception:

Reconstruction before recognition?

Segmentation before recognition?

Invariance to what transformations?

Generalization from "small" or "large" learning sets?

Top-down or bottom-up processing?

Mental images: sparse or dense?

COARSE-TO-FINE SCENE ANALYSIS

STRATEGY: *Design the computational process itself rather than distributions or decision boundaries. Use standard learning algorithms to build components to specifications. Natural progressions and properties then emerge:*

- *CTF:* From broad scope with low power to narrow scope with high power.
- *Graded interpretations:* A running commentary, increasing in precision.
- *Focus of attention:* A spatial distribution of processing - density of work - that is highly data-dependent and (hence) non-uniform.

SMALL-SAMPLE COMPUTATIONAL LEARNING

CLAIM: *No advances in computers or statistical learning will overcome the sample-size problem; some organizational framework is needed.*

- $\{(x_i, y_i), i = 1, ..., n\}$: Training set for inductive learning
  - $x_i \in \mathcal{X}$: Measurement or feature vector;
  - $y_i \in \mathcal{Y}$: True label or explanation of x_i.
- **Examples:**
  - $\mathcal{X}$: Acoustical speech signals; $\mathcal{Y}$: transcription into words
  - $\mathcal{X}$: Natural images; $\mathcal{Y}$: semantic description
  - $\mathcal{X}$: Microarray expression data; $\mathcal{Y}$: class labels
- **Common property:** $\frac{n}{|\mathcal{X}| \vee |\mathcal{Y}|} \approx 0$

GRADED INTERPRETATIONS

Example: Recognizing License Plates:

1. *Focus on the plate ("selective attention")*
2. *Apply pre-stored tests for main characters against "background"*
3. *Devise tests, online, to resolve confusions*
4. *Enforce prior knowledge via global optimization*

Example: Analyzing the Hippocampus:

1. *Detect roughly where it is*
2. *Find "landmarks" to initialize intense computation*
3. *Estimate a dense, 3D, template-to-data map, thereby providing a rich geometric and statistical description*

FORMULATION

- $\mathcal{Y}$: A large number of special explanations for data.
- 0: A dominating "background" explanation or class.
- $\mathbf{Y} \in \{0\} \cup \mathcal{Y}$: The true explanation.

Tasks:

- *Classification*: Determine $\mathbf{Y}$;
- *Figure/Ground Separation*: Determine if $\mathbf{Y} = 0$;
- Invariant Detection: Determine a (random) set $\widehat{Y} \subset \mathcal{Y}$ such that $|\widehat{Y}| \ll |\mathcal{Y}|$ and $P(\mathbf{Y} \in \{0\} \cup \widehat{Y}) \approx 1$

FORMULATION (cont)

Basic Assumption: *There are natural groupings* $A \subset \mathcal{Y}$, *such as similar shapes and "writer", which represent partial explanations.*

Hypothesis Testing: Given $A, B \subset \{0\} \cup \mathcal{Y}$, test

$$\mathbf{Y} \in A \;\; vs. \;\; \mathbf{Y} \in B.$$

Let $X_{AB} \in \{0, 1\}$ denote the data-driven decision.

- **Noncontextual:** $B = A^c$ (nonspecific alternative). Write X_A.
- **Power:** $\beta(X_A) \doteq P(X_A = 0 | \mathbf{Y} \notin A)$. Write $X_{A,\beta}$.
- **Invariance:** For every test: $P(X_{A,\beta} = 1 | \mathbf{Y} \in A) \approx 1$.

FORMULATION (cont)

GOAL: Determine $\widehat{Y}$ based on exploring a *hierarchy*

$$\mathcal{X} = \{X_{A,\beta}, A \in \mathcal{A}, \beta \in [0,1]\}.$$

in a sequential and adaptive manner.

- **Detections** $\widehat{Y}$: Explanations not ruled out by any *performed* test $X_{A,\beta}$:

$$\widehat{Y} = \mathcal{Y} \setminus \bigcup \{A_i, i = 1, ..., K : X_{A_i,\beta_i} = 0\}$$

- **Hierarchical Structure:**
  - *Levels of resolution:* $\mathcal{A} = \bigcup_{l=1}^{L} \mathcal{A}_l$ (disjoint)
  - $\{\mathcal{A}_l\}$: Nested partitions of $\mathcal{Y}$.

MACHINE VISION EXPERIMENTS

EXAMPLE 1: Detect frontal views of highly visible faces in a greyscale image.

- **Face Presentation:** Characterized by geometric pose alone:
  - Position u : Midpoint between the eyes;
  - Scale σ: Distance in pixels between the eyes, assuming $\sigma \geq 10$;
  - Tilt ϕ : Obvious angle.
- **Reference Cell:** Let W be a 16×16 reference window of pixels. $\mathcal{Y}$: A fine partition of $W \times [10, 20] \times [-20°, 20°]$.
- **Pose Decomposition:** Recursively partition $\mathcal{Y}$. Results in a tree-structured hierarchy of pose cells $\mathcal{A} = \{A_{lk}, k = 1, ..., n_l, l = 1, 2, ..., L\}$.

- **Test Construction:** Build test X_A for each $A \in \mathcal{A}$ from training data.
- **Scene Parsing:**
  - **Parallel component:** Visit non-overlapping 16×16 windows and determine $\widehat{Y}$ for surrounding data; downsample, repeat ...
  - **Serial component:** Explore $\mathcal{A}$ breadth-first CTF.

EXAMPLE 2: Detect rectangles amidst clutter. $\mathcal{Y}$: Similar to face detection. (Joint with F. Jung.)

PROBLEM 3: Read the symbols (letters and numerals) on license plates based on close-range photographs of cars. (Joint with Y. Amit.)

- **Symbol Presentation:** $\mathcal{Y} = \{class, \;\; font, \;\; pose\}$.
- **Class/Font/Pose Decomposition:** Recursively partition as before....

COMPUTATION

Strategy: Adaptive (tree-structured) testing procedure:

- $t \in T^o \longrightarrow X_{A_t,\beta_t}$
- $t \in \partial T \longrightarrow \widehat{Y}(t)$, the surviving explanations after testing.

Cost of Testing: The sum of the costs before reaching a decision:

$$C_{test}(T) = \sum_{t \in \partial T} I_{H_t} \sum_{s \downarrow t} c(X_{A_s,\beta_s})$$

where H_t is the event node t is reached. Hence

$$EC_{test}(T) = \sum_{s \in T^o} c(X_{A_s,\beta_s}) P(H_s) = \sum_{A,\beta} c(X_{A,\beta}) q_{A,\beta}(T)$$

where $q_{A,\beta}(T)$ is the probability of performing test $X_{A,\beta}$ in T.

Total Computation: $E\left[C_{test}(T) + c^* |\widehat{Y}(T)|\right]$.

OPTIMIZATION

Under what assumptions are the (sequential testing) strategies which minimize total computation CTF, meaning:

- $(|A| \downarrow)$: A monotonic decrease in scope.
- $(\beta \uparrow)$: A monotonic increase in power.

Two Fundamental Assumptions:

- **Background domination:** Take $P = P_0 = P(.|\mathbf{Y} = 0)$ for measuring power and mean computation.
- **Conditional independence:** $\{X_{A_1,\beta_1}, ..., X_{A_k,\beta_k}\}$ are independent under P_0 whenever $A_1, ..., A_k \in \mathcal{A}$ distinct.

FIXED POWERS

$$\mathcal{X} = \{X_A, A \in \mathcal{A}\},\ c(A) = \ cost,\ \ \beta(A) = \ power$$

THEOREM: (G. Blanchard/DG) *CTF is optimal if*

$$\forall A \in \mathcal{A},\ \frac{c(A)}{\beta(A)} \leq \sum_{B \in \mathcal{C}(A)} \frac{c(B)}{\beta(B)}$$

where $\mathcal{C}(A) =$ *direct children of* A *in* $\mathcal{A}$. *In particular,* $(|A| \downarrow)$ and $(\beta \uparrow)$.

- Each terminal $A \in \mathcal{A}$ has a virtual child with a perfect test of cost c^*.
- For a depth two hierarchy ($\{A_1, B_1, B_2\}$), a n.a.s.c. is

$$\frac{c(A_1)}{\beta(A_1)} \leq \min\left(\frac{c(B_1)}{\beta(B_1)\beta(B_2)} + \frac{c(B_2)}{\beta(B_2)}, \frac{c(B_1)}{\beta(B_1)} + \frac{c(B_2)}{\beta(B_1)\beta(B_2)}\right).$$

FIXED POWERS (cont)

Realistic cost model:

$$c(A, \beta) = \Gamma(|A|) \times \Psi(\beta)$$

where Γ is subadditive ($\Gamma(1) = 1$) and Ψ is convex, increasing ($\Psi(0) = 0$).

COROLLARY: *If power increases with depth and $c(A, \beta)$ is as above, then CTF is optimal.*

Remark: $P_0(\widehat{Y}(T) \neq \emptyset)$ (false positive error) is the nonextinction probability for a non-homogeneous Branching process.

IDEA OF THE PROOF

- Let (CF) be the following property:

  For any subhierarchy, any optimal strategy does X_{A_1} *first.*

- Then (CF) follows from the "magic formula":

$$E_0 C(T) = \sum_{Z \in \mathcal{Z}} P_0(\mathcal{X}_0(T) = Z) \sum_{A \in Z} \frac{c(A)}{\beta(A)}.$$

  where $\mathcal{Z}$ is the set of *coverings* of the (extended) hierarchy and $\mathcal{X}_0(T)$ is the set of tested A's with $X_A = 0$.

- Develop a recursion based on the "projection" of a strategy.

Surprising Equivalence: *Paying* $c(A)$ *for every test performed is the same, on average, as paying* $\frac{c(A)}{\beta(A)}$ *for every null answer and nothing otherwise.*

VARIABLE POWERS

Suppose: $\mathcal{X} = \{X_{A,\beta}, A \in \mathcal{A}, \beta \in [0,1]\}$ and

- $c(X_{A,\beta}) = |A|\Psi(\beta)$ for $\Psi \uparrow$, $\Psi(0) = 0, \Psi(1) = 1$ and Ψ convex.
- $A_s \neq A_t$ for s, t along the same branch of T;

THEOREM: (G. Blanchard/DG)

i) In the CTF strategy, power depends only on scope and $(\beta \downarrow)$*;*

ii) CTF is optimal for $\Psi(x) = 2 - 2\sqrt{1-x} - x$.

CONJECTURES: (Simulation-Based)

i) CTF is optimal among Ψ *convex under mild additional assumptions.*

ii) CTF is optimal for (first-order) Markov hierarchies.

VARIABLE POWERS (cont)

Key Tool: Legendre transform: $\Psi^*(x) = \sup_{\beta \in [0,1]}(x\beta - \Psi(\beta))$.

Example: $(2 - 2\sqrt{1-x} - x)^* = (1+x)^{-1}$.

Cost of the CTF Strategy *Let* C_d *denote the average cost of a complete dyadic hierarchy of depth* d. *Then*

$$C_{d+1} = 2C_d - 2^d \Psi^*(\frac{2C_d}{2^d}), \quad d = 1, 2, ...$$

and

$$\frac{C_d}{2^d} \searrow \Psi'(0), \; d \to \infty.$$

In addition, $\beta_d^* \searrow 0$ *where* β_d^* *is the optimal power for the coarsest test.*

Interpretation: *First do tests which are highly invariant, but have low power (and hence are cheap).*

FINAL COMMENTS

- *Thinking about computation at the start of the day appears useful.*
- *Further practical validation would entail:*
  - Extension to fully deformable objects, e.g., CTF detection of a cat.
  - Accommodating a gigantic number of explanations.
  - Facing the "feedback" dilemma: Is "compositionality" really necessary?
- *Open mathematical questions include:*
  - Demonstrating that the context-based division is optimally efficient.
  - Incorporating dependency, e.g., a Markov hierarchy.
  - Proving that the optimal distribution of total error (FP + FN) puts FN=0.