Compositional Models: Complexity of Representation and Inference

A.L. Yuille and R. Mottaghi
UCLA (JHU)
Dept. Brain and Cognitive Engineering,
Korea University.

"Compositional Models and the Fundamental Problem of Vision"?

Hierarchical Models

- One of the hopes, and expectations, of hierarchical models is that they can represent complex structures in terms of compositions of elementary components – *shared parts.*
- This should yield big gains in the complexity of representation and inference.
- *But how can we analyze and quantify this?*

A Fundamental Problem of Vision

- **Complexity:**
- Set of images is almost infinite (Kersten 1987).
- No. of objects is big 30,000 (Biederman 1984).

- *But the human brain can detect objects and understand scenes within 150 msecs.*
- And we want computer vision systems to do the same.

The Fundamental Problem

- This lecture explores this fundamental problem from the perspective of compositional models.
- *Quantify the gains of part sharing a*nd *executive summary. (Recall objects have a hierarchical distributed representation).*
- *(I):* We analyze compositional models and show they can yield exponential gains in efficiency.
- .(II) We perform a similar analysis for a novel parallel implementation of compositional models.
- *(III) Speculations about the Visual Cortex.*

Compositional Models:

- Examples: Graphical Models for Horses and Players.
- *Executive Summary*: High-level nodes encode coarse descriptions of object. E.g. centroid position
- Details (e.g. leg positions) are specified by lower-level nodes.

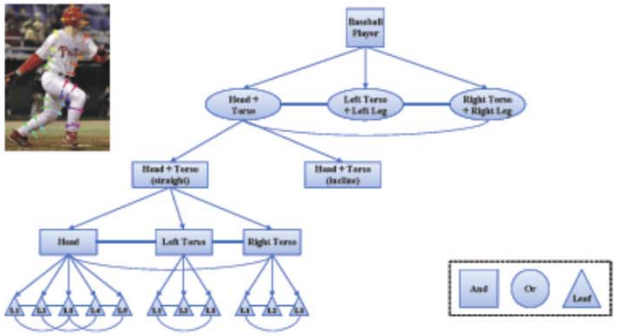
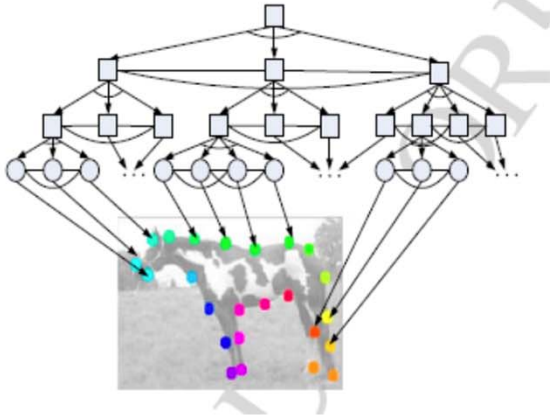
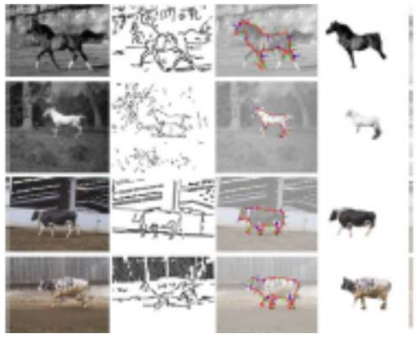


Figure 4. The AND/OR Graph Model (Zhu, Chen, Lin, & Yuille, 2010). The Baseball player is an AND of the head and torso, and left and right legs, but the head is an OR of straight head and torso or an inclined head and torso (top left).





Compositional Model of a Single Object

- Each Object is represented by a graphical model.
- Generative for positions of parts.

$$P(\vec{x}) = P(x_H) \prod_{\nu} P(\vec{x}_{ch(\nu)} | x_\nu; \tau_\nu),$$

- Basic Building Block: Child-Parent Models:

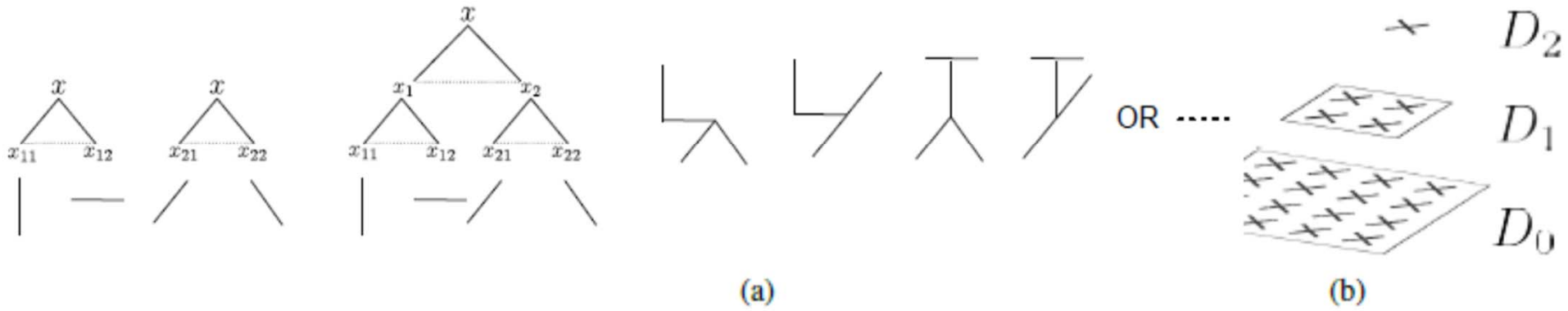
$$\mathbf{P}(\vec{\mathrm{x}}_{ch(\nu)} | \mathrm{x}_\nu, \boldsymbol{\lambda}_\nu) = \boldsymbol{\delta}(\mathrm{x}_\nu - \mathbf{f}(\vec{\mathrm{x}}_{ch(\nu)}))\mathbf{h}(\vec{\mathrm{x}}_\nu; \boldsymbol{\lambda}_\nu)$$

- Generative model for data.

$$P(\mathbf{I} | \{x_l : l \in \mathcal{L}\}) = \prod_{x \in \{x_l\}} P(I(x) | \tau(x)) \times \prod_{x \notin \{x_l\}} P(I(x) | \tau_0),$$

Examples

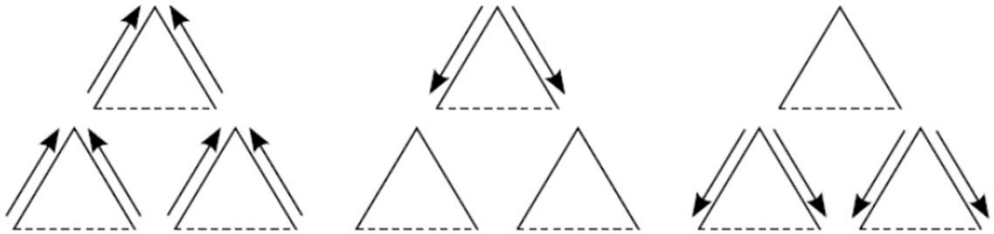
- Left: T's, L's, and their compositions.
- Right: *Executive summary* – quantified by a *Spatial decay factor q* – lower resolution needed for higher-levels of the hierarchy.

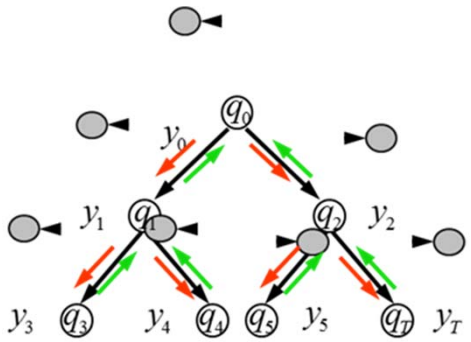


Inference for a Single Object

- For each object, we can perform inference using Dynamic Programming (message passing):
- Bottom-Up and Top-Down pass (cf inside/outside algorithm).

$$\vec{x}^* = \arg\max_{\vec{x}} \{ \sum_{x \in \mathcal{L}} \log \frac{P(I(x)|\tau(x))}{P(I(x)|\tau_0)} + \sum_{\nu} \log P(\vec{x}_{Ch(\nu)}|x_\nu;\tau_\nu) + \log U(x_{\mathcal{H}}) \}.$$





Compositional Inference: Bottom-Up

- DP Example: Level-2 state: $\vec{x} = (x, x_1, x_2, x_{11}, x_{12}, x_{21}, x_{22})$.
- Inference Task is to maximize:

$$\log P(x_1, x_2|x) + \log P(x_{11}, x_{12}|x_1) + \log P(x_{21}, x_{22}|x_2)$$
$$+ \log \frac{P(I(x_{11})|\tau(x_{11}))}{P(I(x_{11})|\tau_0)} + \log \frac{P(I(x_{12})|\tau(x_{12}))}{P(I(x_{12})|\tau_0)} + \log \frac{P(I(x_{21})|\tau(x_{21}))}{P(I(x_{21})|\tau_0)} + \frac{\log P(I(x_{22})|\tau(x_{22}))}{P(I(x_{22})|\tau_0)}.$$

- DP: bottom-up (first step) Computes set

$$\{x_1, \phi(x_1)\} \text{ and } \{x_2, \phi(x_2)\}$$

- By

$$\phi(x_1) = \max_{x_{11}, x_{12}} \{\log P(x_{11}, x_{12}|x_1) + \log \frac{P(I(x_{11})|\tau(x_{11}))}{P(x_{11}|\tau_0)} + \log \frac{P(I(x_{12})|\tau(x_{12}))}{P(I(x_{12})|\tau_0)}$$

$$\phi(x_2) = \max_{x_{21}, x_{22}} \{\log P(x_{21}, x_{22}|x_2) + \log \frac{P(I(x_{21})|\tau(x_{21}))}{P(I(x_{12})|\tau_0)} + \log \frac{P(I(x_{22})|\tau(x_{22}))}{P(I(x_{22})|\tau_0)}$$

- Repeat: $\phi(x) = \max_{x_1, x_2} \{\log P(x_1, x_2|x) + \phi_1(x_1) + \phi_2(x_2)\}$

Compositional Inference: Top-Down

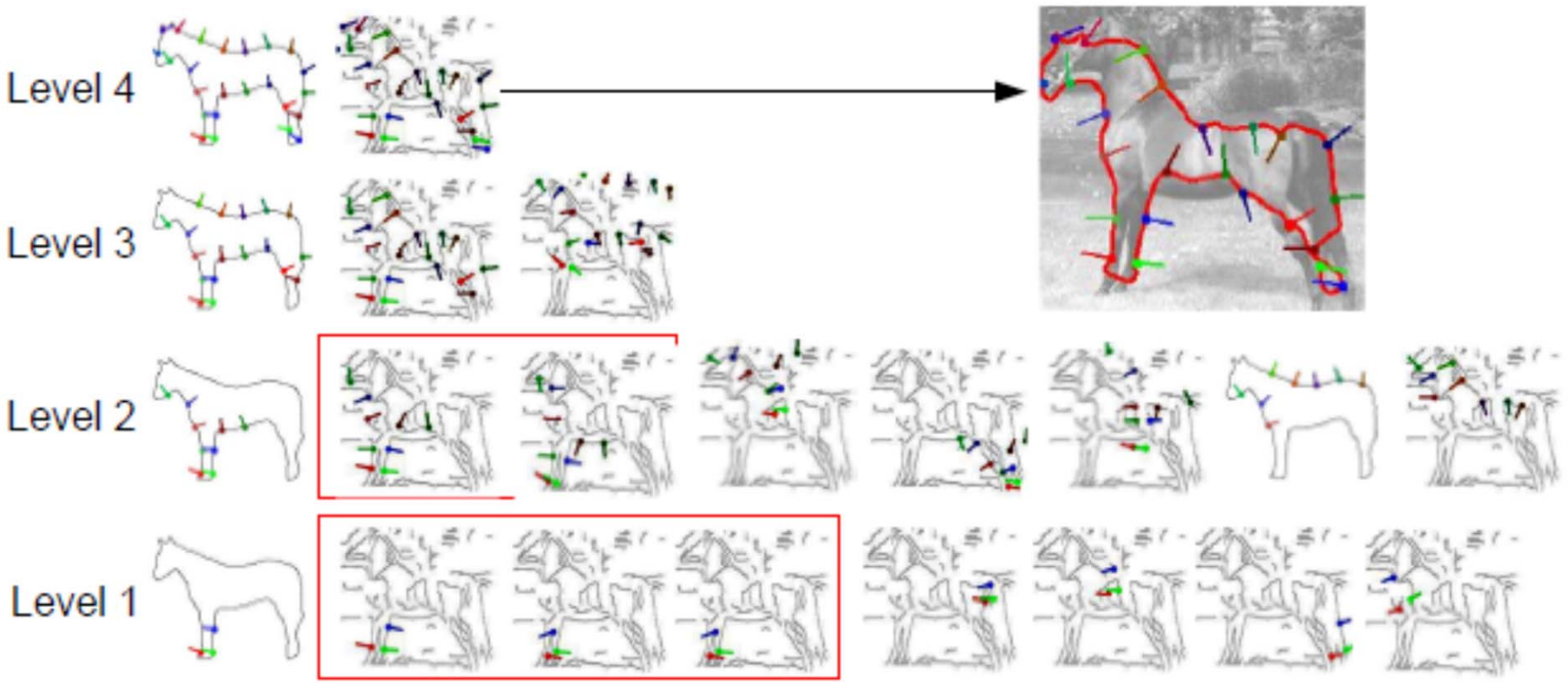
- Top-Down: Estimate $x^* = \arg\max \phi(x).$
- Repeat:

$$(x_1^*, x_2^*) = \arg \max_{(x_1,x_2)} \{\log P(x_1, x_2|x^*) + \phi_1(x_1) + \phi_2(x_2)\}$$

- And so on to obtain: $x_{11}^*, x_{12}^*, x_{21}^*, x_{22}^*.$
- *Intuition:* propagate up hypotheses about the states of subparts of the object. *Increased context as you rise up the hierarchy, less ambiguity.* Estimate coarse structure first --- executive summary. *Top-down uses high-level context to resolve low-levels ambiguities.*

Inference: Illustration

- Bottom-Up



Theories of the Visual Cortex

- Most theories of the visual cortex assume bottom-up/feedforward processing – but some advocate top-down generative approaches.
- Compositional models have aspects of both. They are generative (e.g., synthesis and attention). But allow rapid inference.
- *Inference is done by propagating hypotheses upward in a feedforward pass,* followed by a top-down pass to remove low-level ambiguities.
- *"High-level vision tells low-level to stop gossiping".Murray, Kersten et al.'s fMRI study.*

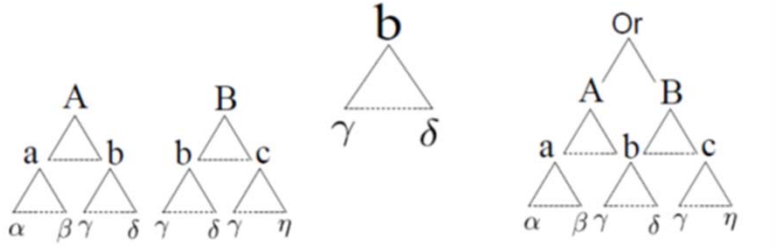
Complexity of Inference for a Single Object

- We can analyze the complexity of inference for a single object – standard analysis of DP.
- Factors:
- (i) No. of Layers -- H.
- (ii) No of children in parent-child --- r.
- (iii) No. of parent-child configurations – C_r
- (iv) Spatial decay factor (ex. summary.) -- q

- Assumed to be the same at all levels of the hierarchy.

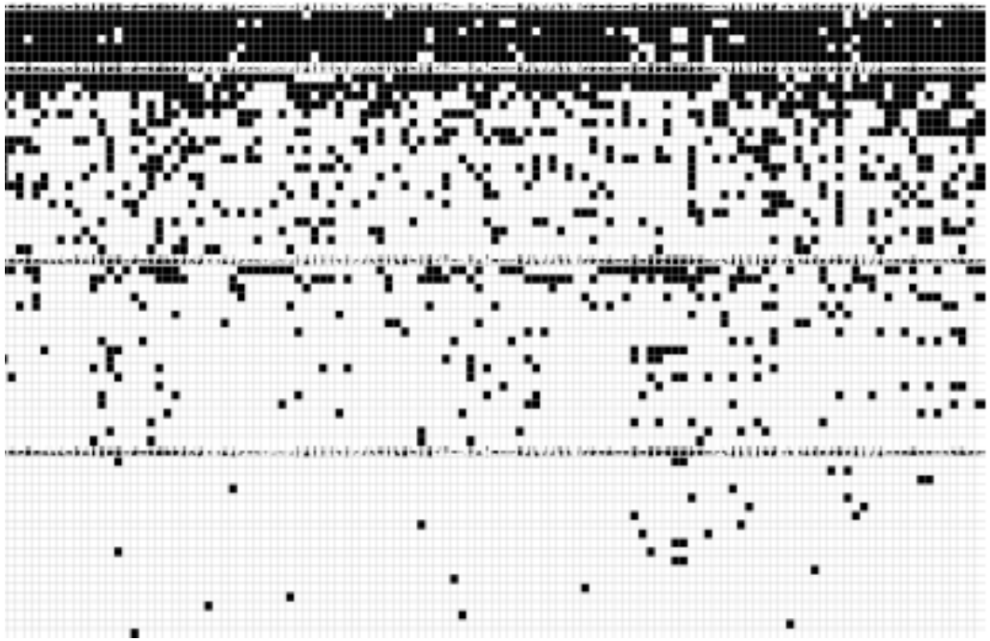
Multiple Objects: Part Sharing

- *If parts and shared between objects we can share the computation between many objects – or many instances of the same object.*
- Captured by hierarchical dictionaries:a,b,c,A,B.
- *Model competition – at top-level – determines which object is present (if any),*
- *No need to train a final classification stage! (Rev.)*



Part Sharing Example: L.Zhu et al. CVPR 2010

- Sharing of parts between 120 objects (horizontal)
- Left: Part Sharing (black)
- Right: Dictionaries – mean shapes only.



Multiple Objects: Inferences

- Inference is performed on the dictionaries with model competition at top-level.
- Recall that a dictionary element at level l is composed (by parent-child relations) of dictionary elements at level l-1
- The complexity of inference depends on the number of dictionary elements.
- Exact inference – relations to UAI work on techniques for speeding up inference on graphs? (E.g., Darwiche and Choi).

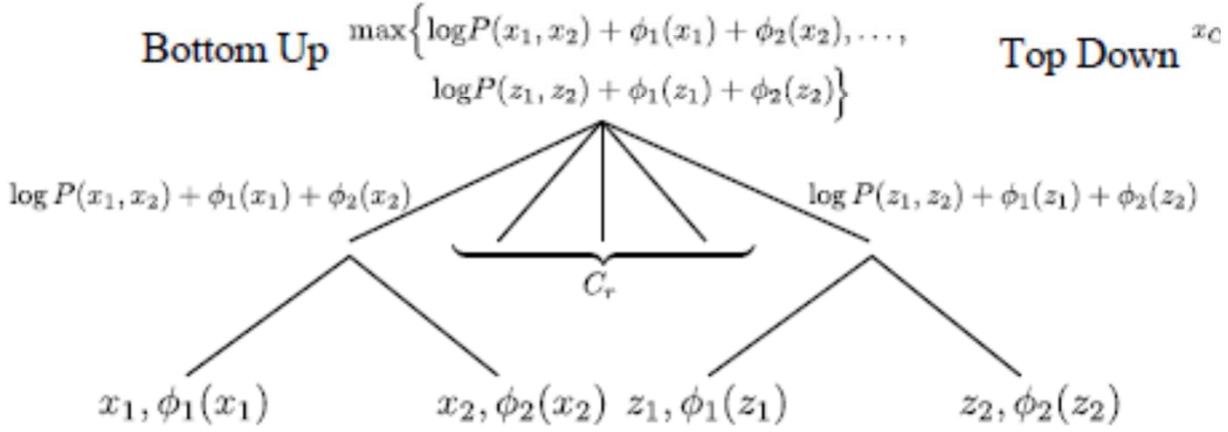
Parallel Implementation.: Convolutional Compositions?

- Dynamic Progamming is naturally parallelized.
- Make copies of the dictionaries at different spatial positions.
- Fewer copies at high-levels (executive summary).
- Non-Linear “Receptive Fields”:

Parallel Implementation of DP

- The bottom-up pass is an AND-like operation followed by an OR-like operation.
- The top-down pass selects the child configuration with maximum score.

$$x_{Ch} = \text{argmax}\Big\{ \log P(x_1, x_2) + \phi_1(x_1) + \phi_2(x_2), \ldots, \log P(z_1, z_2) + \phi_1(z_1) + \phi_2(z_2) \Big\}$$



Complexity for Single Objects.

- The complexity of DP – bottom-up pass is:
- D_0 size of image lattice
- C_r no. child-parent configurations.
- H no. of levels
- r no. of children (e.g. r=3)
- q scale decrease factor (executive summary).

$$N_{bu} = \sum_{h=1}^{\mathcal{H}} |\mathcal{D}_0| C_r r^{\mathcal{H}} (q/r)^h = |\mathcal{D}_0| C_r r^{\mathcal{H}} \sum_{h=1}^{\mathcal{H}} (q/r)^h = |\mathcal{D}_0| C_r \frac{q r^{\mathcal{H}-1}}{1 - q/r} \{1 - (q/r)^{\mathcal{H}}\}.$$

Serial and Parallel Impl. with Part Sharing

- If we do not share parts, then computation scales by the no. M_H of objects.
- For serial Impl. – with part sharing – the complexity depends on the dictionary size M_h at levels h:

$$N_{p_s} = |\mathcal{D}_0| C_r \sum_{h=1}^{\mathcal{H}} |\mathcal{M}_h| q^h.$$

- Parallel Impl – comp. time linear in no. level H.
- But requires no. “neurons”. Copies of dictionaries.
- Trade-off – speed neurons

$$N_n = \sum_{h=1}^{\mathcal{H}} |\mathcal{M}_h| q^h |\mathcal{D}_0|.$$

Analysis: Inference Regimes

- The complexity gains depends on the no. of shared parts: M_h at level h.
- Three Regimes:
- (i) The exponential growth regime (shape?)
- (ii) The empirical regime (CVPR 2010)
- (iii) The exponential decrease regime (appearance?)

Exponential Growth Regime

- This regime is natural for shapes (at the low levels, at least).
- Dictionary elements at one level can be composed with most other dictionary elements to form the dictionary at the next level.

Result 1: If the number of shared parts scales exponentially by $|\mathcal{M}_h| \propto \frac{1}{q^h}$ then we can perform inference for order $q^{\mathcal{H}}$ objects using part sharing in time linear in $\mathcal{H}$, or with a number of neurons linear in $\mathcal{H}$ for parallel implementation. By contrast, inference without part-sharing requires exponential complexity.

Empirical Regime

- This regime was learnt by the unsupervised algorithm (**Thursday Talk).** L.. Zhu et al. CVPR 2010.
- Note: similar to the exponential growth regime for the first few levels, then size of dictionaries decays quickly.

Result 2: If $|\mathcal{M}_h|$ grows slower than $1/q^h$ and if $|\mathcal{M}_h| < r^{\mathcal{H}-h}$ then there are gains due to part sharing using serial and parallel computers. This is illustrated in figure (7)(center panel) based on the dictionaries found by unsupervised computational learning [19]. In parallel implementations, computation is linear in $\mathcal{H}$ while requiring a limited number of nodes ("neurons").

3rd Regime: Exponential Decay

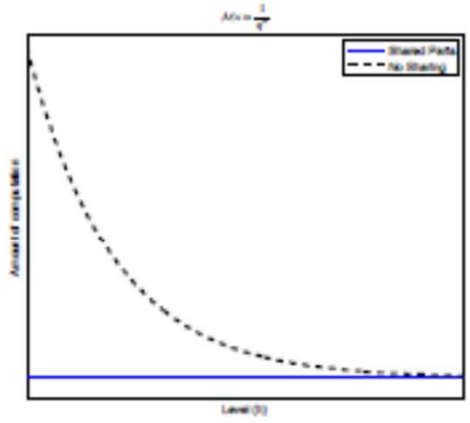
- M_h decreases exponentially with h.
- *This is the "appearance" regime?*
- Intuition: low-level give detailed description:
- (i) Siamese cat fur, (ii) Cat fur, (iii) fur,.
- Executive summary in appearance.

the advantages of parallel computing.

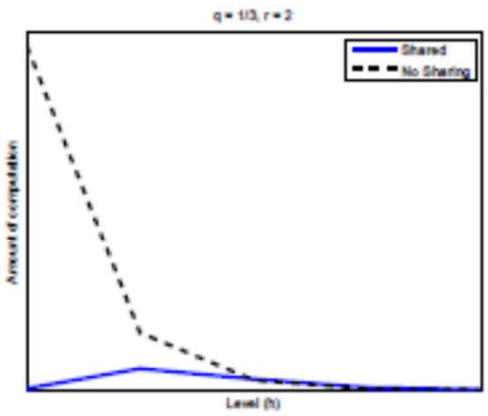
Result 3: If $|\mathcal{M}_h| = r^{\mathcal{H}-h}$ then there is no gain for part sharing if serial computers are used, see figure (7)(right panel). Parallel implementations can do inference in time which is linear in $\mathcal{H}$ but require an exponential number of nodes ("neurons").

Complexity in Figures.

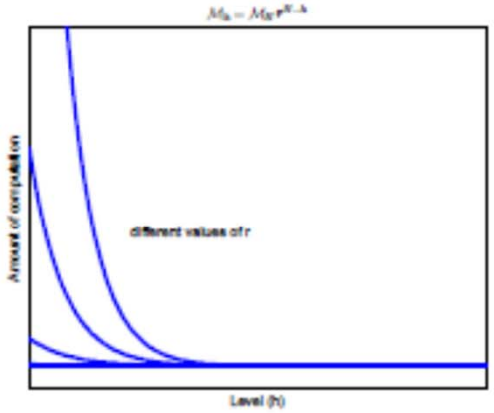
- These illustrate complexity for the three domains.



(a)



(b)



(c)

• Exponential Decay Regime

- This regime is intriguing. It may corresponds to representing the full appearance of objects, and not just their edges.
- Low-level dictionaries represent local appearance patterns.
- In the parallel impl, it requires a very large no. of "neurons" at the lowest levels.
- *Implications for the brain? It suggests that there should be many low-level dictionaries with many local copies.*
- Note: 70% of neurons in the visual cortex are in

Summary

- Complexity Analysis of Compositional Models.
- Serial and Parallel Implementations.
- Gains due to part sharing – compositionality – depend on how the part dictionaries scale with level. Three regimes.
- Visual Cortex speculations: can we derive the structure of the cortex from first principles – as a hierarchical pattern recognition device which is efficient for representation and inference?

References:

- A.L. Yuille and R. Mottaghi. Archive Article.

- L. Zhu et al. Unsupervised Structure Learning. ECCV. 2008.
- L. Zhu et al. Part and Appearance Sharing. CVPR 2010.
- .