Learning the allowable transformations of simple shapes using 3-way interactions

Roland Memisevic, Geoffrey Hinton

University of Toronto

Dec 1, 2007
A common way to capture the statistics of natural images is by learning filters from patches: ICA, PCA, NMF, ...

Bi-partite models explain pixels $y_i$ using hidden “causes” $h_k$. 

What about transformations of natural images?

Can we learn how images change?
Learning data *transformations*

How can we learn *transformations* of data...

...and keep the statistics of the data in mind?

Idea: *Condition* on the input image.

Hidden units now become 'mapping' units that learn to encode transformations.
Restricted Boltzmann machines

(Think of $y$ and $h$ as binary.)

One way to learn (static) filters is by using the *energy*:

$$E(y, h) = - \sum_{jk} w_{jk} y_j h_k$$

... and defining:

$$p(y, h) = \frac{1}{Z} \exp(-E(y, h))$$
Mapping units

- To condition, make the weights a function of the inputs. For example:

\[ w_{jk}(x) = \sum_i w_{ijk}x_i \]

- We get:

\[ E(y, h; x) = -\sum_{ijk} w_{ijk}x_i y_j h_k \]

- Now model the joint conditional over data and hiddens as

\[ p(y, h|x) = \frac{1}{Z(x)} \exp(-E(y, h; x)) \]

\[ Z(x) = \sum_{y,h} \exp(-E(y, h; x)) \]

- (Hinton, Lang 1985), (Sejnowski, 1986), (He, Zemel 2004).
Two views: 1. Input-dependent filters

- Conditional RBM. *Potentials* depend on inputs:
- A CRF, that *learns* its features.
- Input-dependent filters.
Two views: 2. Modulated regression

- Modulated regression. Functions, modulated by hidden units.
- Exponential mixture of experts (with weight sharing).
- Mapping units provide a factorial code for transformations.
Training and inference

Training

▶ For training, maximize $L(W) = \sum_\alpha \log p(y^\alpha|x^\alpha)$

▶ Gradient:

$$\frac{\partial L}{\partial W} = \sum_\alpha \left[ \sum_h p(h|y^\alpha, x^\alpha) \frac{\partial E(h, y^\alpha)}{\partial W} - \sum_{h, y} p(y, h|x^\alpha) \frac{\partial E(h, y)}{\partial W} \right]$$

▶ Contrastive divergence.

Inference

▶ To infer a transformation, compute $p(h|y, x)$. Easy, because of conditional independencies.

▶ Optical flow represented implicitly in the model (as a binary vector). But we can visualize what the model 'thinks' by plotting where each pixel wants to go the most:
Example
Factorial flowfields

Roland Memisevic, Geoffrey Hinton

Learning the allowable transformations of simple shapes using 3-way interactions
Instead of using binary units, it is possible to use any distribution from the **exponential family** as the conditionals for $h$ or $y$.


Can use convolutional version for large images.

Can model nonlinear transformations: $w_{jk}(x) = f_{jk}(x)$, and do feature extraction first.
Flowfields on natural images

- Receptive fields become *conditional receptive fields*.
- Learning on few hours of television broadcast (van Hateren):
Modeling “optical flow”

- Hidden units learn correlation patterns in $x$ and $y$.
- Spatial pooling facilitates generalization and noise suppression:
Learning an invariant metric

▶ We are learning a non-linear manifold *implicitly*: As the set of transformations that generate it.
▶ After training the model, we can define an invariant metric as follows:

"Transformation metric":

▶ To measure the dissimilarity \(d(x, y)\) between new cases \(x\) and \(y\):

1. Set \(\hat{h} = \arg\max_h p(h|x, y)\)
2. Set \(\hat{y} = \arg\max_y p(y|x, \hat{h})\)
3. Set \(d(x, y) = \|y - \hat{y}\|\)

▶ One-step-reconstruction error. ("CD at test-time")
▶ No knowledge included!
Some nearest neighbors
After learning a metric, we can use it, for example, to do classification.
Super-resolution

Learning the allowable transformations of simple shapes using 3-way interactions
Super-resolution

Roland Memisevic, Geoffrey Hinton

Learning the allowable transformations of simple shapes using 3-way interactions
Discriminative de-noising

- If we know how to *corrupt* images, we can learn how to de-corrupt them.
Discussion and further applications

- The task of modeling transformations is inherently “three-way”.
- Here, unlike bi-linear models, only one group of latent variables.
- Learning/inference therefore straightforward.
- Biology?
- Other potential applications: Transfer learning, Learning to cluster, Temporal de-noising, Video compression, Stereo, Depth...