Project Proposal: a Model for Visual Perception as Inference

Alvin Zhang

January 2020

1 Desiderata

A hierarchical generative model for natural video which incorporates:

- A distributed, efficient (sparse) representation, which allows information propagation throughout the hierarchy ~~via top-down, bottom-up, and lateral connections~~,
- Dynamic routing, with a generative model for the routing weights, and
- A top-level representation which exhibits temporal smoothness.

2 Proposed Implementation

Consider the 2-layer hierarchical model for a 1-D signal in Figure 1 (the extension to n-layers and 2-D signals[1] is straightforward but would not be illustrative here).

[1] The only modification when moving to the 2-D case is to additionally allow for 2-D rotation of the Gaussian routing attention maps, which will need to be parameterized in terms of σ^x and σ^y, as well as the correlation coefficient ρ.

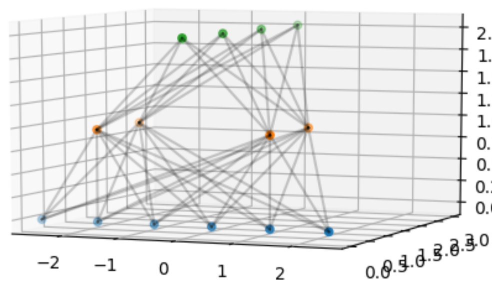

Figure 1: A 2-layer hierarchical model for a 1-D signal, with $N_0 = 6$, $K_0 = 1$, $\alpha_1 = 3$, $N_1 = 2$, $K_1 = 2$, $\alpha_2 = 2$, $N_2 = 1$, $K_2 = 4$.

Notation:

- The i_{th} layer of the model has N_i spatial indices and Φ_i channels.
- The scaling factor from layer i to layer $i-1$ is denoted by $\alpha_i := \frac{N_i}{N_{i-1}}$.
- The lowest layer, layer 0, simply consists of the 1-D signal input.
- Vector v_i is located in the i_{th} layer of the model.
- ϕ_i is the "philter bank"/dictionary for neurons in layer i, and has dimensions $N_i \times \Phi_i \times N_{i-1} \times \Phi_{i-1}$. The $[j,k,l,m]$ entry of ϕ_i is denoted by $\phi_i^{j,k}[l,m]$. In the convolutional setting, $\phi_i^0 = \phi_i^1 = \cdots = \phi_i^{N_i}$.
- a_i are the activity coefficients for each of the filters in ϕ_i, and has dimensions $N_i \times \Phi_i$. The $[j,k]$ entry of a_i is denoted by $a_i^{j,k}$
- μ_i, σ_i are coefficients which are used to route activity between layers i and $i-1$. They have the same dimension and indexing scheme as a_i.
- ϵ_i is the residual between x_i and layer x_{i+1}'s model of x^i. It has the same dimension and indexing scheme as a_i. Unlike previous work, this residual is considered to be a part of the model's (lossless) representation of the image and not an artefact of noise. This allows it to be interpreted in terms of an efficient/sparse coding scheme.

Then the generative model is as follows:

$$a_0^i = \epsilon_0^i + \sum_{j=0}^{N_1} \sum_{k=0}^{\Phi_1} a_1^{j,k} \phi_1^{j,k}[i - \alpha_1 j] \exp\Big(- \frac{(i - \alpha_1 j - \mu_1^{j,k})_2}{2(\sigma_1^{j,k})^2} \Big)$$

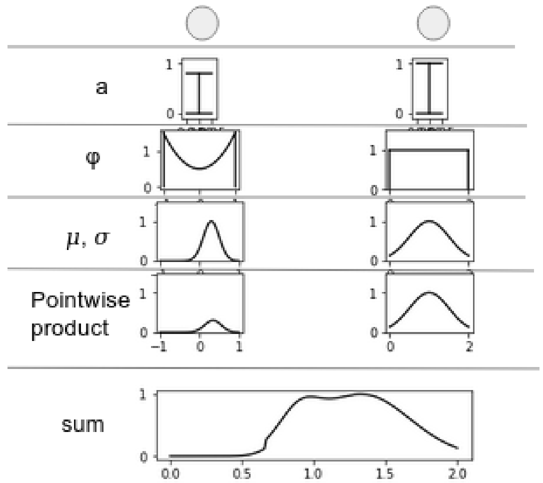

Figure 2: Illustration of how μ and σ can be used to modulate dictionary elements.

Similarly, we model a_1, μ_1, and σ_1 by defining:

$$x_1^i = \epsilon_1^i(x_1) + \sum_{j=0}^{N_2} \sum_{k=0}^{\Phi_2(x_1)} a_2^{j,k}(x_1)\phi_2^{j,k}(x_1)[i - \alpha_2 j]$$
$$\cdot \exp\Big(- \frac{(i - \alpha_2 j - \mu_2^{j,k}(x_1))_2}{2(\sigma_2^{j,k}(x_1))^2} \Big)$$

for $x_1 \in \{a_1, \mu_1, \sigma_1\}$.

This is a hierarchical model not only of the "filter bank"/dictionary activations but also of the attention/dynamic routing patterns.

By treating the ϵ_i residuals as a part of the model, they can be incorporated into the efficient/sparse coding paradigm. Thus we can think of inference as finding a_i, μ_i, and σ_i to minimize $\lambda_0||\epsilon_0||_1+\lambda_1||\epsilon_1||_1+\lambda_2(||a_2||_1+||\mu_2||_1+||\sigma_2||_1)$. This can be done via alternating gradient descent on a_i, μ_i, and σ_i. Note that by setting $\lambda_2 < \lambda_1 < \lambda_0$, higher levels of the hierarchy have increased representational burden placed on them. That is, the model prefers to activate higher-level units over lower-level ones to explain an image.

If the network is shown a video, then it is possible to additionally incorporate a temporal smoothness term $||a_2(t)-a_2(t-1)||_1+||\mu_2(t)-\mu_2(t-1)||_1+||\sigma_2(t)-\sigma_2(t-1)||_1$ into the inference procedure.

Lastly, the "filter bank"/dictionary ϕ^i can be trained by performing inference on a_i, μ_i, and σ_i, and then doing gradient descent on the objective while holding the a_i, μ_i, and σ_i's constant.

3 Considerations

Model log-activities of a_i, μ_i, and σ_i as in Cadieu?

When optimizing a_i, μ_i, and σ_i, is alternating gradient descent good enough, or is something fancier like LCA required?

How far can this be scaled if training on 1 GPU is to take a relatively short time?

4 Sources

http://www.rctn.org/bruno/papers/cadieu-olshausen-nc12.pdf

http://www.rctn.org/vs265/olshausen-etal93.pdf

https://escholarship.org/content/qt1wz289gt/qt1wz289gt.pdf?t=pwzt9dv=lg

https://pdfs.semanticscholar.org/bb00/42b5e48feff89a95182c63bc400c6e6662fe.pdf

https://cs.nyu.edu/ fergus/papers/zeilerECCV2014.pdf