LETTER

Communicated by Michael Lewisky

Sparse Coding via Thresholding and Local Competition in Neural Circuits

Christopher J. Rozell
rozell@gatech.edu
Don H. Johnson
dhj@rice.edu
Richard G. Baraniuk
rich@rice.edu
Department of Electrical and Computer Engineering, Rice University, Houston, TX 77251-1892, U.S.A.

Bruno A. Olshausen
boushause@berkeley.edu
Helen Wills Neuroscience Institute and School of Optometry, University of California, Berkeley, Berkeley, CA 94720-3192, U.S.A.

Motivation

• We want a neurally-plausible coding mechanism
• The world is a dynamic video instead of a static image. How to deal with time-varying stimuli?
Locally Competitive Algorithms (LCA)

- Designed to solve a family of sparse approximation problems
- From the viewpoint of dynamic systems
  - Ordinary differential equation (ODE)
  - e.g. \( \frac{dx}{dt} = f(x(t)) \)
- Similar to neural circuits
- Smooth in time

Sparse Approximation

- Stimulus \( s \in R^N \)
- Dictionary \( \{\phi_m\} \)
  - \( m \in \{1, \ldots, M\}, M > N \)
  - \( \phi_m \in R^N \)
- Optimal
  - \( \min_{a} \|a\|_1 \), s.t. \( s = \sum_{m=1}^{M} a_m \phi_m \)

- NP-hard

- Basis pursuit (BP)
  - \( \min_{a} \|a\|_1 \), s.t. \( s = \sum_{m=1}^{M} a_m \phi_m \)
- Basis pursuit denoising (BPDN)
  - \( \min_{a} (\|s - \sum_{m=1}^{M} a_m \phi_m\|^2 + \lambda \|a\|_1) \)
- Matching pursuit (MP)
  - Initial a residue \( r_0 = s \)
  - Find \( \phi_k \) closest to \( r_0 \)
  - Update to \( r_1 = r_0 - \phi_k \cdot < r_0, \phi_k > \)
Architecture

\[ u(t+1) = \frac{1}{\tau} [b(t) - u(t) - \sum_{n \neq m} G_{m,n} a_n(t)] \]

\[ E(t) = \frac{1}{2} \| s(t) - \hat{s}(t) \|^2 + \lambda \sum_{m} C(a_m(t)) \]

\[ \frac{dE}{da_m} = -b_m + \left( \sum_{n \neq m} G_{m,n} a_n \right) + a_m + \lambda \frac{dC(a_m(t))}{da_m} \]

\[ \lambda \frac{dC(a_m(t))}{da_m} = u_m - a_m \]

\[ \text{Cost function} \]

\[ \text{Thresholding function} \]

\[ a_m(t) = T_j(u(t)) \]

\[ \text{Input: } s(t) \]

\[ \text{a neuron with internal state } u_m(t) \]

\[ \phi_m \text{ a word } \phi_m \text{ in dictionary} \]

\[ \text{leaky current } -u_m(t) \]

\[ \text{excitatory input current} \]

\[ b_m(t) = \langle \phi_m, s(t) \rangle \]

\[ \text{output signal(coefficient)} \]

\[ a_m = T_j(u_m) \]

\[ \text{Inhibition} \]

\[ a_m G_{m,n} \]

\[ G_{m,n} = \langle \phi_m, \phi_n \rangle \]

\[ \hat{s}(t) = \phi_a(t) \]

\[ \phi_l: \text{element of dictionary} \]

\[ u_m: \text{internal state} \]

\[ b_m: \text{input current} \]

\[ a_m: \text{output signal} \]

\[ s(t): \text{input signal} \]

\[ C(\cdot): \text{cost function} \]

\[ \lambda: \text{cost weight} \]
Thresholding activation function

\[ T_{(a,\gamma,\lambda)}(u_m) = \frac{u_m - a\lambda}{1 + e^{-\gamma(u_m - \lambda)}} \]

Additive adjustment

Speed of threshold transition

\[ C_{(a,\gamma,\lambda)}(a_m) = \frac{(1 - a)^2\lambda}{2} + \alpha |a_m| \]

\[ |a_m| \in [0, (1 - \alpha)\lambda] \]

Soft-Thresholding LCA (SLCA)

- \[ C_{(\alpha=1,\infty,\lambda)}(a_m) = \frac{(1-\alpha)^2\lambda}{2} + \alpha |a_m| = |a_m| \]
- \( L^1 \) norm sparsity
- convex
- Compared with BPDN
  - A natural thresholding function that keeps coefficients identically zero
Hard-Thresholding LCA (HLCA)

- $C_{(\alpha=0,\infty,\lambda)}(a_m) = \frac{(1-\alpha)^2 \lambda}{2} + \alpha |a_m| = \frac{\lambda}{2} I(|a_m| > \lambda)$
- $l^0$-like sparsity
- non-convex
- Compared with MP
  - Still work well at some special cases

Experiments on Images

- Dictionary: 4096 elements from Steerable pyramid
- Image patches: 32 x 32
- $\tau=10$ms (membrane time constant)
- Euler method with a time step of 1ms

Illustration of dictionary
Time evolution

Trade-off between MSE and sparsity
Time-Varying Stimuli

- Rewrite dynamic formula for HLCA
  \[ \dot{u}_m(t) = \begin{cases} \begin{aligned} \frac{1}{\tau} & \left( < \phi_m, s(t) > - \hat{s}(t) \right) & \text{if} \ |u_m| < \lambda \\ \frac{1}{\tau} & \left( < \phi_m, s(t) > - \hat{s}(t) \right) & \text{if} \ |u_m| \geq \lambda \end{aligned} \end{cases} \]
- Inertia that smooths the coefficient sequences

Experiments on Sequences

- Sequence: 144 x 144 pixels
- Frame rate is 30 fps.
- Number of changing coefficient at frame \( n \)
  - \( M_{u(n)} \): a set of activated neurons at frame \( n \)
  - \( |M_{u(n-1)} \oplus M_{u(n)}| \) (elusive OR)
- Ratio of the number of changing coefficient at frame \( n \)
  - \( \frac{|M_{u(n-1)} \oplus M_{u(n)}|}{M_{u(n-1)}} \)
Results

![Graphs showing results]

Conclusion

• LCA is a neurally-plausible dynamical system to solve sparse approximations
  • Stable
  • Comparable sparsity levels to most popular methods
  • Readily suited for neural implementation
  • Demonstrate inertial properties for video sequences