Subspace Locally Competitive Algorithms

Dylan M. Paiton
Vision Science Graduate Group
Redwood Center for Theoretical Neuroscience
University of California, Berkeley
Berkeley, California, USA

Steven Shepard
Vision Science Graduate Group
Redwood Center for Theoretical Neuroscience
University of California, Berkeley
Berkeley, California, USA

Kwan Ho Ryan Chan
Department of Mathematics
Redwood Center for Theoretical Neuroscience
University of California, Berkeley
Berkeley, California, USA

Bruno A. Olshausen
Vision Science Graduate Group
Helen Wills Neuroscience Institute
Redwood Center for Theoretical Neuroscience
University of California, Berkeley
Berkeley, California, USA

ABSTRACT

We introduce subspace locally competitive algorithms (SLCAs), a family of novel network architectures for modeling latent representations of natural signals with group sparse structure. SLCA first layer neurons are derived from locally competitive algorithms, which produce responses and learn representations that are well matched to both the linear and non-linear properties observed in simple cells in layer 4 of primary visual cortex (area V1). SLCA incorporates a second layer of neurons which produce approximately invariant responses to signal variations that are linear in their corresponding subspaces, such as phase shifts, resembling a hallmark characteristic of complex cells in V1. We provide a practical analysis of training parameter settings, explore the features and invariances learned, and finally compare the model to single-layer sparse coding and to independent subspace analysis.

KEYWORDS

subspace image coding, group sparse coding, sparse coding, neural network architectures, unsupervised learning, invariance

ACM Reference Format:
Dylan M. Paiton, Steven Shepard, Kwan Ho Ryan Chan, and Bruno A. Olshausen. 2020. Subspace Locally Competitive Algorithms. In *Neuro-inspired Computational Elements Workshop (NICE '20), March 17–20, 2020, Heidelberg, Germany*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3381755.3381765

1 INTRODUCTION

Natural images are well known to have statistical structure that can be efficiently modeled as sparse combinations of oriented, localized, and bandpass filters [Field 1999]. A variety of neural network models have been developed that learn such representations from natural scenes [For example, Aharon et al. 2006; Bell and Sejnowski 1997; Olshausen and Field 1996]. Additionally, the learned representations and responses produced by some of these models have high correspondence with both the linear [Bell and Sejnowski 1997; Olshausen and Field 1997] and non-linear [Zhu and Rozell 2013] response properties of simple cells in area V1 of the primate cortex. Encouraging output neuron activations to be sparse or independent has been a key computational strategy to learn these interesting and efficient representations. Sparse activity of neural firing has been observed experimentally in real biological systems [Vinje and Gallant 2000], and metabolic restrictions may be interpreted as the physical implementation of sparsity constraints in these systems.

Despite encouraging independence for the output neuron activations in these models, many dependencies among responses still remain after model training in the form of common amplitude fluctuations [Hyvärinen and Hoyer 2000; Schwartz and Simoncelli 2001; Wainwright et al. 2001a]. Previous studies have modeled this dependency using a hierarchy of neurons [Hyvärinen and Hoyer 2000; Karklin and Lewicki 2005] that produce comparable outputs to biological V1 simple and complex cells. These networks utilize linear first layer neurons, which limits their likeness to biological neurons as well as their representation capacity and efficiency [Eichhorn et al. 2009; Vilankar and Field 2017]. Other studies have mitigated the dependency using single-layer networks with an adaptive prior that models non-uniformities in the magnitudes among coefficients [Charles et al. 2011; Garrigues and Olshausen 2010; Wainwright et al. 2001b], although these models have not been shown to produce invariant complex cell-like outputs. There is a strong similarity between these two approaches, as a non-uniform prior on first layer coefficients can be implemented via feedback from a hierarchical layer [Lee and Mumford 2003]. Here we present a hierarchical model that uses non-linear first layer neurons and a uniform sparsity prior on second-layer neurons. Our network is an extension of the sparse coding framework that learns subspaces of dependent features for natural images. Information is propagated between first-layer and second-layer neurons during inference, such that a simple uniform prior imposed on the second layer influences the first layer representations and learned weights. We demonstrate that the model is able to produce complex cell like outputs that have partial invariance to important image factors

NICE '20, March 17–20, 2020, Heidelberg, Germany

ACM ISBN 978-1-4503-7718-8/20/03…$15.00
https://doi.org/10.1145/3381755.3381765

such as phase and orientation. Although we do not claim to provide a complete model of complex cell function, we believe adapting the subspace coding framework to sparse coding networks is a positive step towards constructing hierarchical invariant representations of natural scenes, which we identify as a goal of the vision system.

Our primary contribution is a novel architecture for modeling the group sparse structure of natural signals. The architecture is extended from Locally Competitive Algorithms (LCAs), which can be implemented efficiently with analog hardware and thus has significance for neuromorphic computing [Rozell et al. 2008]. We describe subspace locally competitive algorithms for producing two-layer feature-invariant encodings of natural visual scenes. We provide detailed analysis of the invariances learned from natural image data and the coding properties for video inputs. Finally, we compare the model to LCAs as well as independent subspace analysis. Subspace locally competitive algorithms were first introduced in [Paiton 2019], although here we provide more in-depth analyses and comparisons.

2 SUBSPACE LOCALLY COMPETITIVE ALGORITHMS

We will first outline the mathematical details of Subspace Locally Competitive Algorithms (SLCAs), a two-layer extension of Locally Competitive Algorithms (LCAs) [Rozell et al. 2008]. Then we will provide a characterization of the effect of the user-defined sparsity and group-size parameters for training a particular SLCA network. Finally, we will explore the features and invariances learned by the SLCA network to better understand the complex cell-like behavior of the second layer neurons. In our experiments we focus on a single instantiation of SLCAs, although it is possible to explore other versions by modifying the underlying LCA framework as was explored by Rozell et al. [2008] and Charles et al. [2011].

2.1 Model description

In sparse coding, it is typical to first assume a linear generative model:

$$s = \Phi a + \nu = \sum_{k=1}^{L} a_k \Phi_k + \nu, \tag{1}$$

where $s \in \mathbb{R}^P$ is an input image vector and $\Phi \in \mathbb{R}^{P \times L}$ is a learned overcomplete (i.e. $L > P$) dictionary matrix. It is assumed that the input includes small unstructured noise, $\nu \sim \mathcal{N}(0, \Sigma)$, which is drawn from a Gaussian distribution with mean 0 and diagonal variance Σ. The neuron activations are represented with the vector $a \in \mathbb{R}^L$. For a given dictionary, it is necessary to infer a to reconstruct an input signal, s. We also impose a constraint that a is sparse, which encourages conditional independence and efficiency. This is accomplished by minimizing the following energy function:

$$\underset{a}{\operatorname{argmin}} \left(E_{\text{sc}} = \tfrac{1}{2} \|s - \Phi a\|_2^2 + \lambda \sum_{k=1}^{L} C(a_k) \right), \tag{2}$$

where $||\cdot||_2^2$ indicates the squared l_2 norm, which is derived from assuming Gaussian noise, and λ is a sparsity trade-off parameter. For maximum sparsity, the ideal cost function, $C(\cdot)$, would be an l_0 pseudo-norm, which is a direct cost on the number of active units. However in practice it is common to use the l_1 cost – a more computationally tractable relaxation that corresponds to using the absolute value of the activations, $C(a_k) = |a_k|$, and is derived from assuming a Laplacian prior uniformly over the coefficients (for a more thorough probabilistic treatment of sparse coding see [Lewicki and Olshausen 1999]). One of the key features of the LCA model is the ability to easily implement a variety of costs by modifying the threshold function (see [Charles et al. 2011; Rozell et al. 2008] for alternatives and equation 10 for our implementation). For this study we will use a threshold function resembling that used for the l_1 cost, although we identify the exploration of alternative functions as an interesting direction for future study.

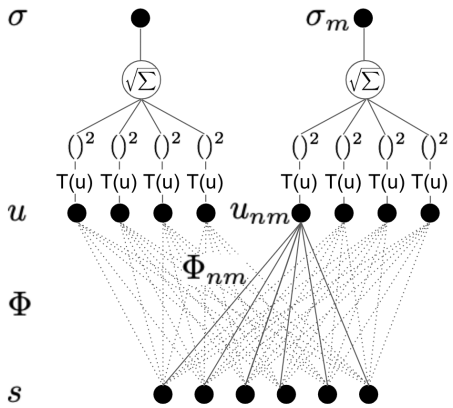

Figure 1: The subspace locally competitive algorithm network. Neurons are grouped such that they learn subspaces of co-active units. Once a group amplitude passes the threshold, all neurons in the group contribute to the output activity. For clarity we omit the all-to-all lateral connections among the first layer neurons, specified by the matrix G in equation (9).

Our SLCA network has two layers. The first layer is composed of sparse coding neurons as in traditional LCA. In the second layer, units pool responses from select groups of neurons in the first layer. The diagram in fig 1 illustrates this network architecture. For SLCA, we reshape the dictionary matrix to be $P \times N \times M$, where P is the number of pixels, N is the number of neurons in a group, and M is the number of groups. Additionally, the activations are reshaped to be $N \times M$. Both N and M are user-defined parameters in our model and are set such that each group has an equal number of neurons and neurons do not belong to more than one group (i.e. groups do not overlap). When comparing to a single-layer sparse coding model, we require that $L = N * M$. The second layer neuron outputs, σ are trained to span correlated subspaces of the input data. Their amplitude is the L_2 norm of the within-group first layer activity vector, a_m:

$$\sigma_m = ||a_m||_2 = \sqrt{\sum_{i=1}^{N} a_{im}^2}, \tag{3}$$

where m indexes the group and $\sum_{i=1}^{N}$ sums over the neurons within the group. This formulation for the group amplitude is akin to the common model of computation for complex cells that is thought

to support their invariance to transformations of certain stimuli features (e.g. [Adelson and Bergen 1985; Heeger 1992]). For a given group, the group amplitude may be equivalent for different combinations of first layer neuron activities. So in this way, the group amplitude can be invariant to specific signal transformations that result in different yet equivalent combinations of first layer neuron activities . Each of these equivalent combinations can be thought of as a direction for a vector in the group subspace. We define this as a unit-length steering vector, z_m, that has the same number of elements as our group activity vector:

$$z_{nm} = \frac{a_{nm}}{\sigma_m}. \tag{4}$$

Thus we can now represent our layer one activations in a polar format with an amplitude σ_m and an angle defined as that between the positive canonical vector pointing from the origin along the horizontal axis and z_m. Additionally, we frame the amplitude units as pooling units akin to complex cells with measurable invariances.

Now we will use these terms to define the SLCA energy function:

$$\begin{aligned} E_{\text{slca}} =& \tfrac{1}{2}||s - \sum_{i=1}^{N}\sum_{j=1}^{M} \sigma_j z_{ij} \Phi_{ij}||_2^2 + \lambda \sum_{j=1}^{M} \sigma_j + \\ & \beta \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{l=1}^{M} \left|\Phi_i^\top \Phi_i - \mathbf{I}_N\right|_{jl}, \end{aligned} \tag{5}$$

where λ and β are user-defined regularization trade-off multipliers, Φ_j is a $P \times N$ matrix representing the basis functions in group j, and $\mathbf{I}_N$ is the $N \times N$ identity matrix. The middle term puts a penalty on the number of active groups for any given input signal, which will encourage independence between group representations. The right most term pressures the within-group weights to be orthogonal, which prevents the pathological solution of within-group neurons learning to have identical weights. This regularization also reduces competition between neurons within groups, which is scaled by the inner product between neighboring neurons' weight vectors. As in sparse coding, we will infer the first layer neuron activations by following the negative energy gradient with respect to that neuron. An interim step for this derivation leads us to an alternative definition of the group direction vector:

$$\begin{aligned} \frac{\partial \sum_{j=1}^{M} \sigma_j}{\partial a_{nm}} &= \frac{\partial \sigma_m}{\partial a_{nm}} \\ &= \frac{a_{nm}}{\sqrt{\sum_{i=1}^{N} a_{im}^2}} \\ &= z_{nm}. \end{aligned} \tag{6}$$

Thus, the negative gradient with respect to activity, a_{nm}, is

$$-\frac{\partial E_{\text{slca}}}{\partial a_{nm}} = \sum_{h=1}^{P} s_h \Phi_{hnm} - \sum_{i=1}^{N}\sum_{j=1}^{M} G_{ijnm} a_{ij} - \lambda z_{nm}, \tag{7}$$

where $G_{ijnm} = \sum_{h=1}^{P} \Phi_{hij}\Phi_{hnm}$ is the dictionary Gramian tensor.

As was done for LCA networks, our first layer neurons will maintain a state variables, u, with dynamics that are governed by the energy function. Neurons produce activations when their states exceed a threshold, which we represent as a rate code, a. Superthreshold neurons will inhibit the other first layer neurons in the network through horizontal connections, defined by the tensor G. To derive the governing equations, we first group the self inhibition terms from equation 7 and assign them to the neuron's internal state, u:

$$\begin{aligned} u_{nm} \coloneqq f_\lambda(a_{nm}) &= a_{nm} + \lambda z_{nm} \\ &= (\sigma_m + \lambda)\, z_{nm} \\ &= a_{nm}(1 + \frac{\lambda}{\sigma_m}). \end{aligned} \tag{8}$$

In order for the network dynamics to settle to a minimum of the energy function, we assign the update rule for each neuron's internal state to be proportional to equation 7, giving us

$$\tau \dot{u}_{nm} = b_{nm} - \sum_{ij \neq nm} G_{ijnm} a_{ij} - u_{nm}, \tag{9}$$

where $b_{nm} = \sum_{h=1}^{P} s_h \Phi_{hnm} = \Phi_{nm}^\top s$ is the neuron's feedforward drive and τ is the neuron's membrane time constant. The SLCA network differs from LCA networks in that all neurons within a group will produce outputs when that group exceeds the threshold. However, as long as a_m and u_m are related by a monotonically increasing function, the network activations will settle to a global minima [Rozell et al. 2008]. The output amplitude in terms of the group threshold $a_{nm} = T_\lambda(u_{nm})$ is

$$T_\lambda(u_{nm}) \coloneqq f_\lambda^{-1}(u_{nm}) = \begin{cases} 0, & ||u_m||_2 \leq \lambda \\ (||u_m||_2 - \lambda)\dfrac{u_{nm}}{||u_m||_2}, & ||u_m||_2 > \lambda. \end{cases} \tag{10}$$

This formulation is equivalent to an independently derived threshold function for an LCA implementation used to solve sparse signal approximation problems with a block-l_1 cost function [Charles et al. 2011].

Deriving the learning process for SLCA follows the same procedure as for the original LCA, although the modified energy function results in an additional term. The basis functions, Φ_{nm}, are optimized by performing gradient descent on E_{slca} while fixing the coefficient values, a, computed from the SLCA inference step. This yields the update rule

$$\begin{aligned} \Delta\Phi_{pnm} =& \eta \Bigg[\left(s_p - \sum_{i=1}^{N}\sum_{j=1}^{M} \sigma_j z_{ij} \Phi_{pij} \right) a_{nm} + \\ & 2\beta \sum_{j=1}^{M} \Phi_{pnj} \text{sign}\left(\sum_{h=1}^{P} \Phi_{hnj}\Phi_{hnm} - \delta_{jm} \right) \Bigg], \end{aligned} \tag{11}$$

In practice, we used the automatic gradient computation provided by the TensorFlow library [Abadi et al. 2015] for updating the network weights.

The resulting network produces non-linear neurons at both the first and second layer in the hierarchy. Like the LCA network, this network can also be implemented in analog circuit hardware that settles to the energy minimum [Charles et al. 2011; Rozell et al. 2008]. In the next section, we will present and analyze the features learned when training the SLCA network on natural image patches.

2.2 Features learned

All models were trained on the van Hateren natural scenes dataset [Hateren and Schaaf 1998]. We preprocessed the data by transforming the pixel values to log intensity, whitening, normalizing the

images to have zero mean and unit standard deviation, and finally extracting 16 by 16 pixel patches. Image whitening was done using an approximate Fourier method on the whole images [See section 5.9.3 of Hyvärinen et al. 2009], where we first performed a 2D Fourier transform on the image, then multiplied it by a whitening filter, and finally performed an inverse Fourier transform. The whitening filter was composed by multiplying together a ramp (that has a slope of 1 and rises with frequency) component and a low-pass (starting at 0.7 times the Nyquist frequency) component.

When trained on natural images, the SLCA weights learn to tile orientations, spatial frequencies, and positions. They also learn to have within-group similarities, such as equal orientation or position. Figure 2 shows the results of a parameter sweep to convey the effect of group size on the model performance. From this process we selected a lambda value of 1.0 and group size of 4 for the rest of the experiments.

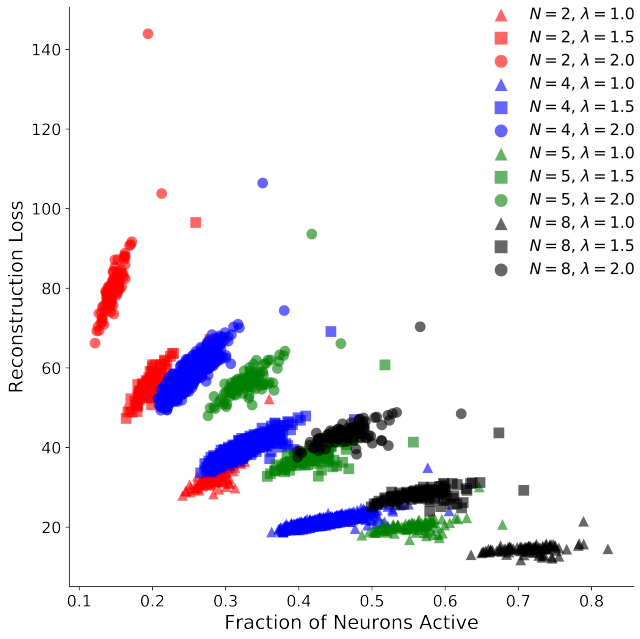

Figure 2: Reconstruction error versus sparsity for group sizes of M: 2, 4, 5, 8, with 1280 neurons (5x overcomplete) and $\beta = 0.2$. For each group size, we tested three values for λ: 1.0, 1.5, and 2.0, where lower values will result in a higher fraction active. Moreover, higher group size leads to a lower fraction of neurons active on average.

Figure 3 shows a selection of the features learned with the SLCA network. Although the group structure constrains the dictionary, the network still learns to tile spatial frequencies, positions, orientations, and phases.

2.3 Qualitative assessment of cell invariance

A given second layer neuron can represent a continuous variation of features that are defined by the space spanned by the first layer weight vectors. We illustrate this in figure 4 by generating

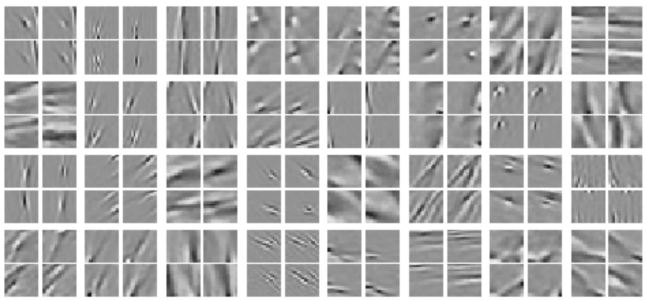

Figure 3: Weights learned for our chosen parameter set, $N = 4$, $\lambda = 1.5$, and $\beta = 0.2$. SLCA converges to features that are like LCA, but grouped to have similar properties within groups, and different properties across groups.

images while holding group amplitude constant and varying group direction. To generate the images, we compute angle vectors, z_m, from natural images that evoke a large group amplitude, σ_m. Reconstructions are created from the inferred z_m and a one-hot group amplitude vector with $\sigma_m = 1$. Each row of images produce equal group activations from a different second layer neuron.

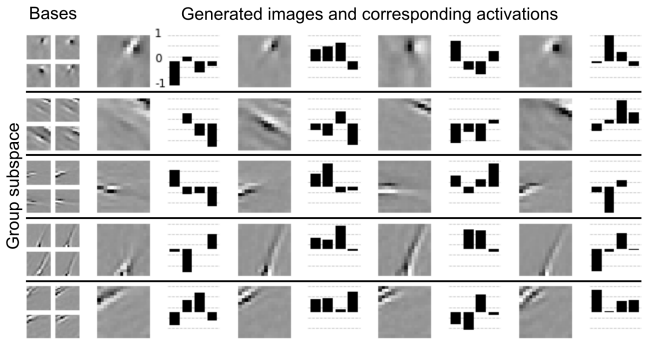

Figure 4: Group neurons have equal response for a variety of image features. Each row corresponds to a different group, or subspace. The leftmost column shows the four layer 1 basis functions that makeup the given group. For the remaining columns we set the group amplitudes to a one-hot vector with $\sigma_m = 1$ and present (left) the reconstructions and (right) the corresponding layer 1 neuron amplitudes (i.e. the z_m vector). The z_m are computed from natural images that evoke a strong group response. The bar charts show the z_{im} value for each neuron weight, where the left to right bars correspond to a clockwise rotation starting in the top left in the basis image grid. We selected groups that have equal amplitudes for images that vary in phase, position, orientation, edge length, and spatial frequency for the descending rows, respectively.

There is likely a continuum of cells exhibiting response properties between what is classically considered "simple" and "complex", as visually responsive cells in V1 exhibit a variety of shared non-linear response properties that can be observed with different stimulus variations [De Valois et al. 1982; Dean and Tolhurst 1983;

Mechler and Ringach 2002]. Characterizing cell responses to controlled stimulus, such as sinusoidal gratings, is extremely valuable in understanding their basic response properties, but limited in its ability to define the high-dimensional response geometry of individual cells. An alternative, complementary, approach is to map out the regions in stimulus space where the neuron's responses are equal – their iso-response surfaces [Golden et al. 2016]. In practice, this would be difficult to do for biological neurons due to the high dimensionality of the input and large amount of stimulus variations required, however it is tractable to characterize for model neurons. In order to better understand the response properties of our second layer neurons, we visualized their iso-response contours, which are found when an iso-response surface of a neuron is projected onto a two-dimensional subspace.

It is possible for neuron iso-response contours to be straight or curved. Curved contours may bend towards the origin (endo-origin) or away from it (exo-origin). Linear and pointwise non-linear neurons (i.e. any model with rectified, sigmoid, hyperbolic tangent, or functionally similar non-linearities) will produce straight contours [Golden et al. 2016]. Exo-origin curvature is indicative of selectivity, meaning the neuron will respond to a smaller set of possible stimuli than a neuron with straight iso-response contours [Vilankar and Field 2017]. This form of curvature can result from AND-like operators, gain control, divisive normalization, and other competitive computations, such as that found among sparse coding neurons. Conversely, endo-origin curvature is indicative of invariance, where the neuron responds equally to variations between the vectors defining the image plane [Golden et al. 2016]. This form of curvature can result from OR-like operators and computations implemented in classic energy models. As an illustrative example, consider a simple two-layer network that has two first layer neurons with π-phase-shifted weights and a single pooling neuron. The first layer of neurons produce linear outputs, $a = W^\top s$. The pooling neuron uses SLCA grouping actions as described before, producing output that is the square-root of the sum of the squares of the within-group first layer neurons' activities, $\sigma = \sqrt{\sum_i a_i^2}$. This will result in invariance to phase modulation. To visualize the iso-response contours, we first construct a stimulus set from a two-dimensional plane (or cross-section) in stimulus space that is defined by the two orthogonal weight vectors from the first layer. Each individual point in this two-dimensional plane can be represented as a stimulus in image signal space, and thus a discrete sampling of the points can be collected to form a stimulus set. Next we compute the second layer's output for each image in our stimulus set. We visualize the iso-response contours by coloring the points in the two-dimensional plane according to the second layer neuron's normalized response and then binning the values with fixed bin widths. The resulting bin boundaries are iso-response contours. For this example network, the iso-response contours of the second layer neurons will be concentric circles with centers at the origin. Of course, this basic complex cell model is insufficient to explain the variety of response properties observed in biological neurons. More complicated models will not have perfectly circular iso-response contours and will likely have both endo- and exo-origin curvature when viewing different projections of their high-dimensional response surface. A neuron that has both types of curvature indicates invariance along some pair of dimensions (e.g. those which explore phase variation) and selectivity along another pair of dimensions (e.g. those which explore position variation) [Golden et al. 2016]. In figure 5 we show SLCA complex cell iso-response contours in two-dimensional cross-sections of the P-dimensional image space. In each plot the horizontal axis is defined by a weight vector that corresponds to a "target" layer 1 neuron in some group. A second layer 1 "comparison" neuron is chosen to either be in the same group (figure 5, left column) or a different group (figure 5, right column). The vertical axis for the image plane is then chosen using a single step of the Gram-Schmidt process starting from the target neuron and comparison neuron pair. The plot color values correspond to the second layer neuron that contains the target first layer neuron in its group. The second layer neurons can exhibit both endo- and exo-origin curvature, indicating a nuanced relationship between the neuron's selectivity and invariance properties.

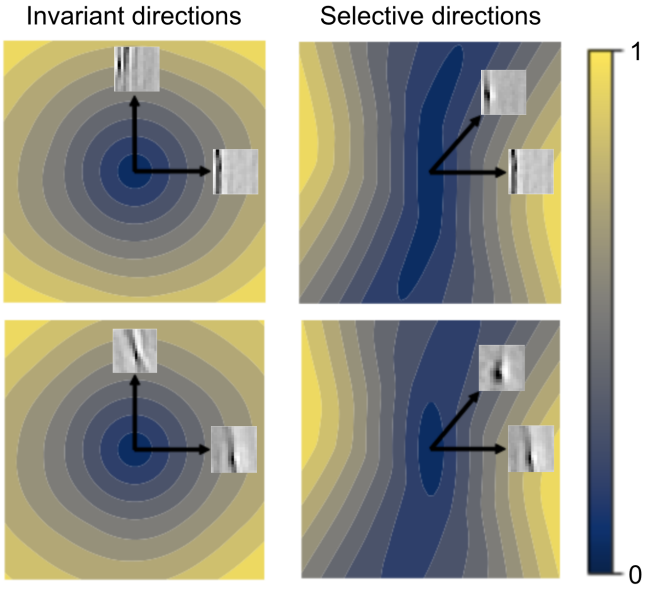

Figure 5: SLCA second layer cells have both endo- and exo-origin iso-response curvature. Contour lines indicate regions of equal response (iso-response contours) for second layer cells. Each row visualizes outputs from a different second-layer cell. Image planes are spanned by two first layer neuron basis functions that are either within the same group (left column) or in different groups (right column). Endo-origin curvature for within-group planes indicates that the second layer cell is partially invariant to linear combinations of the two defining vectors. Exo-origin curvature for different-group planes indicates that the second layer cell is selective to differences between the two defining features, such as orientation or position.

3 COMPARISONS TO SPARSE CODING AND INDEPENDENT SUBSPACE ANALYSIS

Another successful natural scene model is independent components analysis (ICA) [Bell and Sejnowski 1997; Hyvärinen 1999],

which was extended by [Hyvärinen and Hoyer 2000] to independent subspace analysis (ISA). Both models represent images as linear combinations of elements from a complete dictionary, $a = \Phi^\top s$. The process for learning Φ encourages that the elements in the a vector are statistically independent. However, the encoding process itself is linear and thus all elements in Φ are likely to have some inner-product with s and the code produced will likely not have any zero-valued outputs. Like SLCA, the ISA extension incorporates a pooling layer whose outputs are computed as the square-root of the sum-of-squares of a pre-defined group of inputs from the first layer. ISA pooling layer units are shown to exhibit phase and shift invariant properties, much like mammalian V1 complex cells. However, there are two key differences between ISA and SLCA regarding the representation and encoding procedures. One difference is that the first layer representation in SLCA is overcomplete. The second key difference is that the first layer encoding process for SLCA is non-linear. These two differences result in improved efficiency as well as selectivity [Eichhorn et al. 2009; Lewicki and Sejnowski 2000; Vilankar and Field 2017].

In figure 6, we characterize the weights learned by computing their peak spatial position, frequency, and orientation using the feature's 2-D Fourier transform. Because SLCA is based on the LCA framework, we are able to train an overcomplete model and learn a finer sampling of these natural variations in the image data. The group constraint on the LCA model does not restrict it from tiling the different variation parameters. However, SLCA does learn more weights with lower spatial frequencies and centered positions when compared to LCA.

The natural variations from frame to frame in video sequences are small, and therefore the representation of adjacent frames should also be small. The LCA network has previously been shown to provide increased stability for encoding video inputs [Rozell et al. 2008] when compared to alternative sparse coding methods. To test how this influences the pooling unit outputs we encoded a natural video with the LCA, ISA, and SLCA models. The video is from a head-mounted GoPro 5 camera recording at 90 frames-per-second while the wearer walked around the UC Berkeley campus. In figure 7 we show improved stability in the second layer of the SLCA model when compared to ISA. Instead of an explicit slowness prior [for example, as was done in Lies et al. 2014; Wiskott and Sejnowski 2002], SLCA's slowness comes from the LCA network and it is achieved because the differential equation driving the network neurons has hysteresis with respect to the changing input.

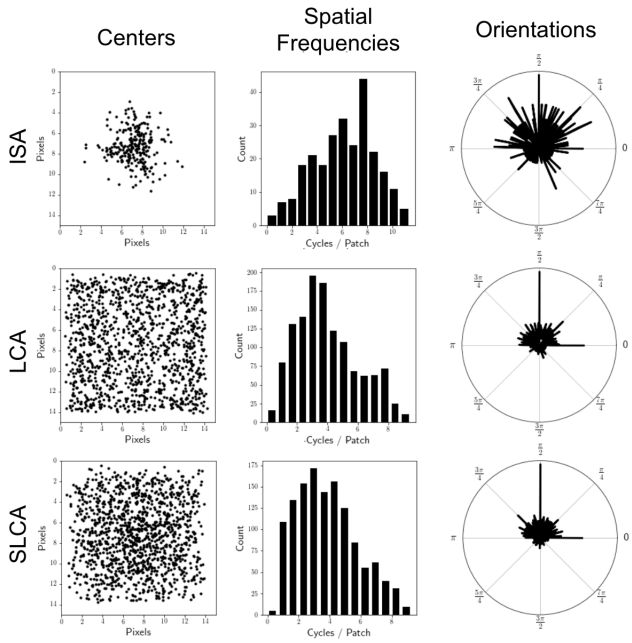

Figure 6: Features learned with ISA, LCA, and SLCA. The LCA and SLCA models are both 5 times overcomplete, while the ISA model is complete. From left to right, the columns are the spatial positions, spatial frequencies, and orientations of each function. For the left column, points indicate the center of individual basis functions learned for each model.

4 DISCUSSION

The linear responses of natural images to Gabor-like filters have kurtotic histograms centered around zero, indicating that they provide for an efficient coding strategy [Field 1999]. The responses also have strong dependencies in the form of common amplitude fluctuations, which can be observed by viewing the joint histograms of individual filters [Wainwright et al. 2001a]. A group sparse prior can provide a more flexible and efficient representation that resolves much of these dependencies [Garrigues and Olshausen 2010]. Importantly, neurons in networks that utilize a group sparse prior exhibit response properties that have been commonly used to identify complex cells in biological vision systems [Hyvärinen and Hoyer 2000; Pollen and Ronner 1983], suggesting that it is a promising method for encoding natural signals. In this study, we presented a novel network architecture for producing nonlinear, decomposed representations of natural images. We demonstrated that the second layer representation jointly encodes feature identity as a group amplitude and qualitative elements of the feature as a group direction. The network is an extension of Locally Competitive Algorithms, which can be implemented efficiently in analog hardware, although we reserve developing an SLCA hardware circuit analogy for future work. We first provided details about the network's training behavior for a variety of parameter settings. Then we explored the learned invariances by visualizing generated images that produce equal second layer responses. Our network neurons have iso-response geometry with both exo-origin curvature, indicating selectivity, and endo-origin curvature, indicating invariance, which was previously hypothesized but never demonstrated [Golden et al. 2016]. Finally, we demonstrate that subspace sparse coding produces more stable video representations than the independent subspace analysis model.

The work herein was focused on laying groundwork for a group sparse coding architecture that extends an analog inference algorithm. There is a relationship between framing subspace coding in terms of a non-uniform prior over a single layer network, as was done in [Garrigues and Olshausen 2010], and imposing a uniform prior on the second layer of a two-layer network, as we have

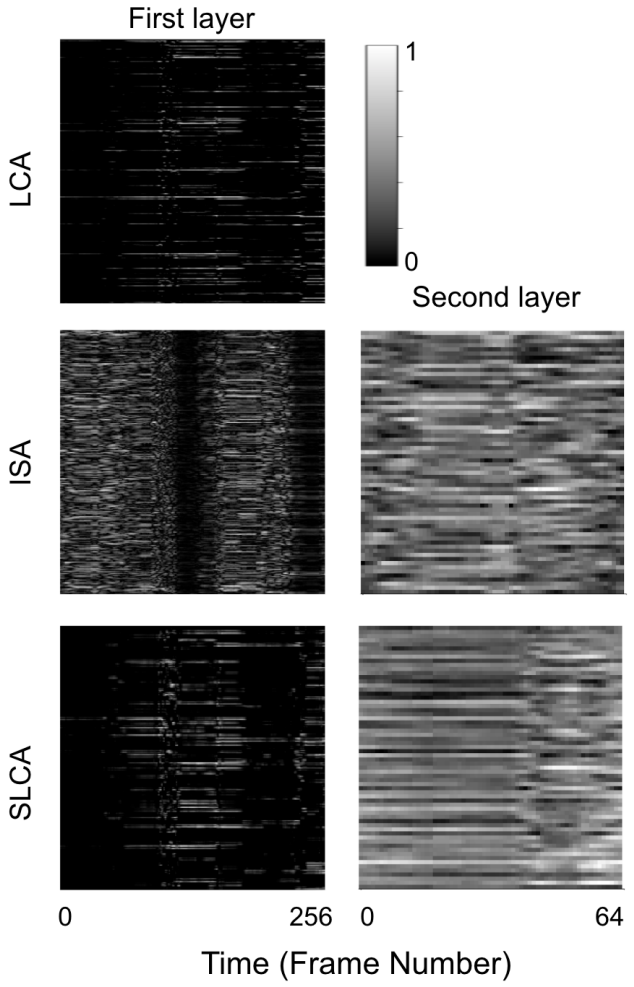

Figure 7: SLCA produces more stable codes for video inputs than ISA. Extending the LCA network to encode for features subspaces results in a more stable code for video inputs, which is an important property for decoding by downstream neurons. To illustrate this effect we show a sampling activations of 256 first layer units and 64 second layer units from each model. Each row in the matrices is a neuron index, while each column is a frame from a video of natural scenes. The grayscale value indicates the normalized neuron activation.

done here. A better understanding of this relationship could help to resolve the current model inconsistency where within-group neurons share their internal states. We are also interested in using the SLCA to learn groupings of data without supervised signals in a semi-supervised object detection task.

ACKNOWLEDGMENTS

We would like to thank Yubei Chen for feedback regarding the weight orthogonality objective and the other members of The Redwood Center for valuable discussions. Additionally, we thank the ENIGMA NSF research program and the Berkeley AI Research group for providing funding for this work.

REFERENCES

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/ Software available from tensorflow.org.

Edward H Adelson and James R Bergen. 1985. Spatiotemporal energy models for the perception of motion. *Josa a* 2, 2 (1985), 284–299.

Michal Aharon, Michael Elad, and Alfred Bruckstein. 2006. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing* 54, 11 (2006), 4311–4322.

Anthony J Bell and Terrence J Sejnowski. 1997. The "independent components" of natural scenes are edge filters. *Vision research* 37, 23 (1997), 3327–3338.

Adam S Charles, Pierre Garrigues, and Christopher J Rozell. 2011. Analog sparse approximation with applications to compressed sensing. *arXiv preprint arXiv:1111.4118* (2011).

Russell L De Valois, Duane G Albrecht, and Lisa G Thorell. 1982. Spatial frequency selectivity of cells in macaque visual cortex. *Vision research* 22, 5 (1982), 545–559.

Andrew F Dean and David J Tolhurst. 1983. On the distinctness of simple and complex cells in the visual cortex of the cat. *The Journal of Physiology* 344, 1 (1983), 305–325.

J Eichhorn, F Sinz, and M Bethge. 2009. Natural Image Coding in V1: How Much Use is Orientation Selectivity? *PLOS Computational Biology* 4 (Oct. 2009). arXiv:q-bio.NC/0810.2872v2

David J Field. 1999. Wavelets, vision and the statistics of natural scenes. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 357, 1760 (1999), 2527–2542.

Pierre Garrigues and Bruno A Olshausen. 2010. Group sparse coding with a laplacian scale mixture prior. In *Advances in neural information processing systems*. 676–684.

J R Golden, K P Vilankar, M C K Wu, and D J Field. 2016. Conjectures regarding the nonlinear geometry of visual neurons. *Vision research* 120 (March 2016), 74–92.

J. H. van Hateren and A. van der Schaaf. 1998. Independent Component Filters of Natural Images Compared with Simple Cells in Primary Visual Cortex. *Proceedings: Biological Sciences* 265, 1394 (Mar 1998), 359–366.

David J Heeger. 1992. Half-squaring in responses of cat striate cells. *Visual neuroscience* 9, 5 (1992), 427–443.

Aapo Hyvärinen. 1999. Fast and robust fixed-point algorithms for independent component analysis. *IEEE transactions on Neural Networks* 10, 3 (1999), 626–634.

Aapo Hyvärinen and Patrik O Hoyer. 2000. Emergence of phase-and shift-invariant features by decomposition of natural images into independent feature subspaces. *Neural computation* 12, 7 (2000), 1705–1720.

Aapo Hyvärinen, Jarmo Hurri, and Patrick O Hoyer. 2009. *Natural image statistics: A probabilistic approach to early computational vision.* Vol. 39. Springer Science & Business Media.

Yan Karklin and Michael S Lewicki. 2005. A hierarchical Bayesian model for learning nonlinear statistical regularities in nonstationary natural signals. *Neural computation* 17, 2 (2005), 397–423.

Tai Sing Lee and David Mumford. 2003. Hierarchical Bayesian inference in the visual cortex. *JOSA A* 20, 7 (2003), 1434–1448.

Michael S Lewicki and Bruno A Olshausen. 1999. Probabilistic framework for the adaptation and comparison of image codes. *JOSA A* 16, 7 (1999), 1587–1601.

Michael S Lewicki and Terrence J Sejnowski. 2000. Learning overcomplete representations. *Neural computation* 12, 2 (2000), 337–365.

Jörn-Philipp Lies, Ralf M Häfner, and Matthias Bethge. 2014. Slowness and sparseness have diverging effects on complex cell learning. *PLoS computational biology* 10, 3 (2014), e1003468.

Ferenc Mechler and Dario L Ringach. 2002. On the classification of simple and complex cells. *Vision research* 42, 8 (2002), 1017–1033.

Bruno A Olshausen and David J Field. 1996. Sparse coding of natural images produces localized, oriented, bandpass receptive fields. *Nature* 381, 60 (1996), 609–619.

Bruno A Olshausen and David J Field. 1997. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision research* 37, 23 (1997), 3311–3325.

Dylan M Paiton. 2019. *Analysis and applications of the Locally Competitive Algorithm.* Ph.D. Dissertation. UC Berkeley.

Daniel A Pollen and Steven F Ronner. 1983. Visual cortical neurons as localized spatial frequency filters. *IEEE Transactions on Systems, Man, and Cybernetics* 5 (1983), 907–916.

Christopher J Rozell, Don H Johnson, Richard G Baraniuk, and Bruno A Olshausen. 2008. Sparse coding via thresholding and local competition in neural circuits.

Neural computation 20, 10 (2008), 2526–2563.

Odelia Schwartz and Eero P Simoncelli. 2001. Natural signal statistics and sensory gain control. *Nature neuroscience* 4, 8 (2001), 819–826.

Kedarnath P Vilankar and David J Field. 2017. Selectivity, hyperselectivity, and the tuning of V1 neurons. *Journal of vision* 17, 9 (2017), 9–31.

William E Vinje and Jack L Gallant. 2000. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science* 287, 5456 (2000), 1273–1276.

Martin J Wainwright, Odelia Schwartz, and Eero P Simoncelli. 2001a. Natural image statistics and divisive normalization: modeling nonlinearity and adaptation in cortical neurons. In *Statistical Theories of the Brain*, Rajesh Rao, Bruno Olshausen, and Michael Lewicki (Eds.). MIT Press, Chapter 10, 266–290.

Martin J Wainwright, Eero P Simoncelli, and Alan S Willsky. 2001b. Random cascades on wavelet trees and their use in analyzing and modeling natural images. *Applied and Computational Harmonic Analysis* 11, 1 (2001), 89–123.

Laurenz Wiskott and Terrence J Sejnowski. 2002. Slow feature analysis: Unsupervised learning of invariances. *Neural computation* 14, 4 (2002), 715–770.

Mengchen Zhu and Christopher J Rozell. 2013. Visual nonclassical receptive field effects emerge from sparse coding in a dynamical system. *PLoS Computational Biology* 9, 8 (2013).