What Are the Invariant Occlusive Components of Image Patches? A Probabilistic Generative Approach

Authored by:

J?rg L?cke
Zhenwen Dai
Georgios Exarchakis

Abstract

We study optimal image encoding based on a generative approach with non-linear feature combinations and explicit position encoding. By far most approaches to unsupervised learning learning of visual features, such as sparse coding or ICA, account for translations by representing the same features at different positions. Some earlier models used a separate encoding of features and their positions to facilitate invariant data encoding and recognition. All probabilistic generative models with explicit position encoding have so far assumed a linear superposition of components to encode image patches. Here, we for the first time apply a model with non-linear feature superposition and explicit position encoding. By avoiding linear superpositions, the studied model represents a closer match to component occlusions which are ubiquitous in natural images. In order to account for occlusions, the non-linear model encodes patches qualitatively very different from linear models by using component representations separated into mask and feature parameters. We first investigated encodings learned by the model using artificial data with mutually occluding components. We find that the model extracts the components, and that it can correctly identify the occlusive components with the hidden variables of the model. On natural image patches, the model learns component masks and features for typical image components. By using reverse correlation, we estimate the receptive fields associated with the model's hidden units. We find many Gabor-like or globular receptive fields as well as fields sensitive to more complex structures. Our results show that probabilistic models that capture occlusions and invariances can be trained efficiently on image patches, and that the resulting encoding represents an alternative model for the neural encoding of images in the primary visual cortex.

1 Paper Body

Probabilistic generative models are used to mathematically formulate the generation process of observed data. Based on a good probabilistic model of the data, we can infer the processes that have generated a given data point, i.e., we can estimate the hidden causes of the generation. These hidden causes are usually the objects we want to infer knowledge about, be it for medical data, biological processes, or sensory data such as acoustic or visual data. However, real data are usually very complex, which makes the formulation of an exact data model infeasible. Image data are a typical example of such complex data. The true generation process of images involves, for instance, different objects with different features at different positions, mutual occlusions, object shades, lighting 1

mask

feature

Translation

Component 1

Component 2

Background

Figure 1: An illustration of the generation process of our model. conditions and reflections due to self-structure and nearby objects. Even if a generative model can capture some of these features, an inversion of the model using Bayes? rule very rapidly becomes analytically and computationally intractable. As a consequence, generative modelers make compromises to allow for trainability and applicability of their generative approaches. Two properties that have, since long, been identified as crucial for models of images are object occlusions [1?5] and the invariance of object identity to translations [6?13]. However, models incorporating both occlusions and invariances suffer from a very pronounced combinatorial complexity. They could, so far, only be trained with very low dimensional hidden spaces [2, 14, 15]. At first glance, occlusion modeling is, furthermore, mathematically more inconvenient. For these reasons, many studies including style and content models [16], other bi-linear models [17, 18], invariant sparse coding [19, 20], or invariant NMF [21] do not model occlusions. Analytical and computation reasons are often explicitly stated as the main motivation for the use of the linear superposition of components (see, e.g., [16, 17]). In this work, we for the first time study the encoding of natural image patches using a model with both non-linear feature combinations and translation invariances.

2

A Generative Model with Non-linear and Invariant Components

The model used to study image patch encoding assumes an exclusive component combination, i.e., for each pixel exclusively one cause is made responsible. It thus shares the property of exclusiveness with visual occlusions. The model will later be shown to capture occluding components. We will, however, not model explicit occlusion using a depth variable (compare [2]) but will focus on the exclusiveness property. The applied model is a novel version of the invariant occlusive components model studied for mid-level vision earlier [22]. We first

briefly reiterate the basic model in the following and discuss the main differences of the new version afterwards. We consider image patches y with D2 observed scalar variables, y = (y1 , . . . , yD2). An image patch is assumed to contain a subset from a set of H components. Each component h can be located at a different position denoted by an index variable xh ? {1, . . . , D2 }, which is associated with a set of permutation matrices that covers all the possible planar translations {T1 , . . . , TD2 } (similar formulations have also been used in sprite models [14, 15]). Each component h is modeled to appear in an image patch with probability ?h ? (0, 1). Following [22], we do not model component presence and absence explicitly but, for mathematical convenience, assign the special ?position? ?1 to all the components which are not chosen to generate the patch. Assuming a uniform distribution for the positions, the prior distribution for components and their positions is thus given by: p(x—?) =

Y h

(1 ? ?h , xh = ?1 p(xh —?h), p(xh —?h) = ?h , , otherwise D2

(1)

where the hidden variable x = (x1 , . . . , xH) contains the information on presence/absence and position of all the image components. In contrast to linear models, the studied approach requires two sets of parameters for the encoding of image components: component masks and component features. Component masks describe where an image component is located, and component features describe what a component encodes (compare [2, 3, 14, 15]). High values of mask parameters ? h encode the pixels most associated with a component h but the encoding has to be understood relative to a global component position. The feature parameters w h encode the values of a component?s features. Fig. 1 shows an example 2

of the mask and feature parameters for two typical low-level visual features. Given a particular position, the mask and feature parameters of the component are transformed to the target position by multiplying a corresponding translation matrix like Txh ? h and Txh w h . When generating an image patch, two or more components may occupy the same pixel, but according to occlusion the pixel value is exclusively determined by only one of them. This exclusiveness is formulated by defining a mask variable m = (m1 , . . . , mD2). For a pixel at a position d, md determines which component is responsible for the pixel value. Therefore, md takes a value from the set of present components ? = {h—xh 6= ?1} plus a special value ?0? indicating background, and the prior distribution of m

? is defined as: 2 ?0 p(m— x, A) =

D Y

p(md — x, A),

p(md = h— x, A) =

d=1

P

h0 ?? (Txh0 (Txh ? h)d

? h0)d

P

h0 ?? (Txh0
? h0)d
??
0+
??
0+
,
h=0
,
h??
,
(2)

where A = (?1 , . . . , ? H) contains the mask parameters for all the components, and ?0 defines the mask parameter for background. The mask variable md chooses the component h with a high likelihood if the translated mask parameter of the corresponding component is high at the position d. Note that md follows a mixture model given the presence/absence and positions of all the components x. This can be thought of as an approximation to the distribution of mask variables marginalizing the depth orderings and pixel transparency in the exact occlusive model (see Supplement A for a comparison). After drawing the values of the hidden variables x and m, an image patch can be generated with a Gaussian noise model, which is given by: (2

p(y —m, x, ?) =
D Y
p(yd —md , x, ?),
p(yd —md = h, x, ?) =
d=1
2), h=0 N (yd ; B, ?B , h)d , ? 2), h ? ? N (yd ; (Txh w
(3)

2) are all the model where ? 2 is the variance of components, and ? = (? , W, A, ? 2 , ?0 , B, ?B 2 . parameters. The background distribution is a Gaussian distribution with mean B and variance ?B Compared to an occlusive model with exact EM (see Supplement A), our approach will use the exclusiveness approximation and a truncated posterior approximation in order to make learning tractable.

The model described in (1) to (3) has been optimized for the encoding of image patches. First, feature variables are scalar to encode light intensities or input by the lateral geniculus nucleus (LGN) rather than color features for mid-level vision. Second, to capture the frequency of presence for individual components, we implement the learning of the prior parameter of presence ? . Third, the pre-selection function in the variational approximation (see below) has been adapted to the usage of scalar valued features. As a scalar value is much less distinctive than the sophisticated image features used in [22], the preselection of components has been changed to the complete component instead of only salient features.

3

Efficient Likelihood Optimization

Given a set of image patches Y = (y (1) , . . . , y (n)), learning is formulated as inferring the best model parameters w.r.t. the log-likelihood L = p(Y —?). Following the Expectation Maximization (EM) approach, the parameter update equations are derived. The equations of the mask parameter ? h , and feature parameter w h are the same as in [22]. Additionally, we derived the update equation for the prior parameter of presence: N ?h =

1 X N n=1

X

p(x— y (n) , ?).

(4)

x?{xh 6=?1}

By learning the prior parameters ?h , the probabilities of individual components? presence can be estimated. This allows us to gain more insights about the statistics of image components. In the update equations, a posterior distribution has been estimated for each data point, which corresponds to the E-step of an EM algorithm. The posterior distribution of our model can be decomposed as: p(m, x— y , ?) = p(x— y , ?)

QD2

d=1

p(md — x, y , ?),

(5)

in which p(x—y , ?) and p(md —x, y , ?) are estimated separately. Computing the exact distribution of p(x—y , ?) is intractable, as it includes the combinatorics of the presence/absence of components and their positions. An efficient posterior approximation, Expectation Truncation (ET), has been successfully employed. ET approximates the posterior distribution as a truncated distribution [23]: p(x— y , ?) ? P

p(y, x—?) , if x ? Kn , p(y, x0 —?) n

(6)

x0 ?K

and zero otherwise. If Kn is chosen to be small but to contain the states with most posterior probability mass, the computation of the posterior distribution becomes tractable while a high accuracy 3

Figure 2: Numerical Experiments on Artificial Data. (a) eight samples of the generated data sets. (b) The parameters of the eight components used to generate the data set. The 1st row contains the binary transparency parameters and the 2nd row shows the feature parameters. (c) The learned model parameters (H = 9). The top plot shows the learned prior probabilities ? . The 1st row shows the mask parameters A; the 2nd row shows the feature parameters W ; the 3rd row gives a good visualization of only the frequent used elements/pixels (setting the feature parameter whd of the elements/pixels with ?hd ¡ 0.5 to zero). (d) The result of inference given a image patch (shown on the left). The right side shows the four components inferred to be present (each takes a column). The 1st and 2nd rows show the mask and features parameters shifted according to the MAP inference xMAP , and the 3rd row shows the inferred posterior p(md

—xMAP , y , ?). All the plots are heat map (Jet color map) visualizations of scalar values. of the approximations can be maintained [23]. To select a proper subspace Kn , ? features (pixel intensities) are chosen according to their mask parameters. Based on the chosen features, a score value S(xh) is computed for each component at each position (see [22]). We select H 0 components, denoted as H, for the candidates that may appear in the given image according to the probability p(y , x ?h —?). x ?h corresponds to the vector x with xh = x?h and the rest components absent 0 (xh0 = ?1, h 6= h), where x?h is the best position of the component h w.r.t. S(xh). This is different from the earlier work [22], where Kn is constructed directly according to S(xh). For each component, we select the set of its candidate positions Xh , xh ? Xh , which contains the p best positions w.r.t. S(xh). Then the truncated subspace Kn is defined as: X X Kn = { x—(sj ? ? and si = 0, ?i ? / H) or sj 0 ? 1},

(7)

j0

j

where sh represents the presence/absence state of the component h (sh = 0 if xh = ?1 ? xh ? / Xh and sh = 1 if xh ? Xh). To avoid converging to local optima, we used the directional annealing scheme [22] for our learning algorithm.

4

Numerical Experiments on Artificial Data

The goal of the experiment on artificial data is to verify that the model and inference method can recover the correct parameters, and to investigate inference on the data generated according to occlusions with explicit depth variable. We generated 4?4 gray-scale image patches. In the data set, eight different components are used, which are four vertical ?bars? and four horizontal ?bars?, and each bar has a different intensity and has a binary vector indicating its ?transparency? (1 for non-transparent and 0 for transparent, see Fig. 2b) . When generating an image patch, a subset of components is selected according to their prior probabilities ?h = 0.25, and the selected components are combined according to a random depth ordering (flat priors on the ordering). A component with smaller depth will occlude the components with larger depth, and for each image patch we sample a new depthordering. For the pixels in which all the selected components are transparent, the value is determined according to the background with zero intensity (B = 0). All the pixels generated by components are subject to a Gaussian noise with ? = 0.02 and the pixels belonging to the background have a Gaussian noise with ?B = 0.001. In total, we generated N = 1, 000 image patches. Fig. 2a shows eight samples. The artificial data is similar to data generated by the occlusive components analysis model (OCA; [2]), except of the use of scalar features and the assumption of shift-invariance. Fig. 2c shows the learned model parameters on the generated data set. We learned nine components (H = 9). The initial feature value W was set to randomly selected data points. The initial mask parameter A was independently and uniformly drawn from the interval (0, 1). The initial annealing temperature was set to T = 5. After keeping constant for 20 iterations,

the temperature linearly decreased to 1 in 100 iterations. For the robustness of learning, ? decreased together with the temperature from 0.2 to 0.02, and an additive Gaussian noise with zero mean and ?w = 0.04 was 4

injected into W and ?w gradually decreased to zero. The algorithm terminated when the temperature was equal to 1 and the difference of the pseudo data log-likelihood of two consecutive iterations was sufficiently small (less than 0.1%). The approximation parameters used in learning was H 0 = 8, ? = 4, p = 2 and ? = 3. In this result, all the eight generative components have been successfully learned. The 2nd to last component (see Fig. 2c) is a dumpy component (low ?h , i.e., very rarely used). Its single pixel structure is therefore an artifact. With the learned parameters, the model could infer the present components, their positions and the pixel-to-component assignment. Fig. 2d shows a typical example. Given an image patch on the left, the present components and their positions are correctly inferred. Furthermore, as shown on the 3rd row, the posterior probabilities of the mask variable p(md —x, y , ?) give a clear assignment of the contributing component for each pixel. This information is potentially very valuable for tasks like parts-based object segmentation or to infer the depth ordering among the components. We assess the reliability of our learning algorithm by repeating the learning procedure with the same configuration but different random parameter initializations. The algorithm recovers all the generative components in 11 out of 20 repetitive runs. The 9 runs not recovering all bars did still recover reasonable solutions with usually 7 bars out of 8 bars represented. In general, optima of bar stimuli seem to have much more pronounced local optima, e.g., compared to image patches.

5

Numerical Experiments on Image Patches

After we verified the inference and learning algorithm on artificial data, it was applied to patches of natural images. As training set we used N = 100, 000 patches of size 16 ? 16 pixels extracted at random positions from random images of the van Hateren natural image database [24]. We modeled the sensitivity of neurons in the LGN using a difference-of-Gaussians (DoG) filter for different positions, i.e., we processed all patches by convolving them with a DoG kernel. Following earlier studies (see [5] for references), the ratio between the standard deviation of the positive and the negative Gaussian was chosen to be 1/3 and the amplitudes chosen to obtain a mean-free centersurround filter. Fig. 3a shows some samples of the image patches after preprocessing. Our algorithm learned H = 100 components from the natural image data set. The model parameters were initialized in the same way as for artificial data. The annealing temperature was initialized with T = 10, kept constant for 10 iterations, the temperature linearly decreased to 1 in 100 iterations. ? decreased together with the temperature from 0.5 to 0.2, and an additive Gaussian noise with zero mean and ?w = 0.2 was injected into W and ?w gradually decreased to zero. The approximation parameters used for learning were H 0 = 6, ? = 4, p = 2 and ? = 50. After 134 iterations, the model parameters had essentially converged. Figs. 3bc show the learned mask parameters and the learned feature values for all the 100 components. Mask parameters define the frequently used areas within

a component, and feature parameters reveal the appearance of a component on image patches. As can be observed, image components are very differently represented from linear models. See the component in Fig. 3d as an example: mask parameters are localized and all positive; feature parameters have positive and negative values across the whole patch. Masks and features can be combined to resemble a familiar Gabor function via point-wise multiplication (see Fig. 3d). All the above shown component representations are sorted in descending order according to the learned prior probabilities of occurrence ? (see Fig. 3e).

6

Estimation of Receptive Fields

For visualization, mask and feature parameters can be combined via point-wise multiplication. To more systematically and quantitatively interpret the learned components and to compare them to biological experimental findings, we estimated the predicted receptive fields (RFs). RFs estimates were computed with reverse correlation based on the model inference results. Reverse correlation can be defined as procedure to find the best linear approximation of the components? presence given h, h ? an image patch y (n) . More formally, we search for a set of predicted receptive fields R {1, . . . , H} that minimize the following cost function: f=

1 N

P P n

x?Kn

p(x — y (n) , ?)

T ? y (n) h (Rh Txh

P

(n)

? sh)2 + ?

P T h Rh Rh ,

(8)

where y is the nth stimulus and ? is the coefficient for L2 regularization. sh is a binary variable representing the presence/absence state of the component h, where sh = 0 if xh = ?1, and sh = 1 5

(a)

(e)

(b)

RF

(c)

(d)

(f)

Figure 3: The invariant occlusive components from natural image patches. (a) shows 20 samples of the pre-processed image patches. (b) shows the mask parameter and (c) shows the feature parameter. (d) shows an example of the relation with the learned model parameters and the estimated RFs. (e) shows the learned prior probabilities ? . (f) shows the estimated Receptive Fields (RF). The RFs were fitted with 2 dimensional Gabor and DoG functions. The dashed line marks the RFs that have a more globular structure. The solid lines

mark the RFs the were fitted accurately by a Gabor function. The dotted lines marks the RFs that were not approximated very well by the fitted function. All the shown model parameters in (b-c) and receptive fields in (f) are sorted in descent according to ? . The plots (a-d) and (f) are heat map visualization with local scaling on individual fields (Jet color map), and (a), (c) and (f) fix light green to be zero. otherwise. As our model allows the components to be at different locations, the reverse correlation is computed by shifting the stimuli according to the inferred location of each components. T?xh represents the transformation matrix applied to the stimulus for the component h, which is the opposite transformation of the inferred transformation Txh (T?xh Txh = 1). For the absent components, the stimulus is used without any transformations (T?1 = 1). Due to the intractability of computing an exact posterior distribution, given a data point, the cost function only sums across the truncated subspace Kn in the variational approximation (see Sec. 3). h can be estimated as: By setting the derivative of the cost function to zero, R

?1 P

(n) ? h = ?N 1 + P hT?x R (Txh y (n))T iqn s(T?xh y (n))T iqn hy n n h (9)

where h?iqn denotes the expectation value w.r.t. the posterior distribution p(x —y (n) , ?) and 1 is h , we often observe that many of the eigenvalues of the data an identity matrix. When solving R PN (n) h covariance matrix n=1 hT?xh y (T?xh y (n))T iqn are close to zero, which makes the solution of R very unstable. Therefore, we introduce a L2 regularization to the cost function. The regularization coefficient ? is chosen between the minimum and maximum element of the data covariance matrix. The estimated receptive fields are not sensitive to the value of the regularization coefficient ? as long as ? is large enough to resolve the numerical instability (see Supplement for a comparison of the receptive fields estimated with different ? values). From the experiments with artificial data and 6

natural image patches, we observed that the L2 regularization successfully eliminated the numerical stability problem. Fig. 3f shows the RFs estimated according to our model. For further analysis, we matched the RFs using Gabor functions and DoG functions as was suggested in [5]. If we factored in the occurrence probabilities, we found that the model considered about 17% of all components of the patches to be globular, 56% to be Gabor-like and 27% to have another structure (see Supplement for details). The prevalence of ?center-on? globular fields may be a consequence of the prevalence of convex object shapes.

7

Discussion

The encoding of image patches investigated in this study separates feature and position information of visual components. Functionally, such an encoding has been found very useful, e.g., for the construction of object recognition systems. Many state-of-the-art systems for visual object classification make use of convolutional neural networks [12, 25, 26]. Such networks compute the responses of a set of filters for all positions in a predefined area and use the maximal response for further processing ([12] for a review). If we identify the

predefined area with one image patch as processed by our approach, then the encoding studied here is to some extent similar to convolutional networks: (A) it uses like convolutional networks one set of component parameters for all positions; and (B) a hidden component variable of the generative model integrates or ?pools? the information across all positions. As the here studied approach is based on a generative data model, the integration across positions can directly be interpreted as inversion of the generation process. Crucially, the inversion can take occlusions of visual features into account while convolutional networks do not model occlusions. Furthermore, the generative model uses a probabilistic encoding, i.e., it assigns probabilities to positions and features of a joint feature and position space. Ambiguous visual input can therefore be represented appropriately. In contrast, convolutional networks use one position for each feature as representation. In this sense a convolutional encoding could be regarded as MAP estimate for the feature position while the generative integration could be interpreted as probabilistic pooling. Many bilinear models have also been applied to image patches, e.g., [17, 18]. Such studies do report that neurally plausible receptive fields (RFs) in the form of Gabor functions emerge [17, 18]. Likewise, invariant versions of NMF [21] or ICA (in the form of ISA [9] have been applied to image patches. In addition to Gabors, we observed in our study a large variety of further types of RFs. Gabor filters with different orientations, phase and frequencies, as well as globular fields and fields with more complex structures (Fig. 3f). Gabors have been studied since several decades, globular and more complex fields have attracted attention in the last couple of years. In particular, globular fields have attracted attention [5, 27, 28] as they have been reported together with Gabors in macaques and other species ([29] and [5] for further references). Such fields have been associated with occlusions before [5, 28, 30]; and our study now for the first time reports globular fields for an occlusive and translation invariant approach. The results may be taken as further evidence of the connection between occlusions and globular fields. However, also linear convolutional approaches have recently reported such fields [19, 31]. Linear approaches seem to require a high degree of overcompleteness or specific priors while globular fields naturally emerge for occlusion-like non-linearities. More concretely: for non-invariant linear sparse coding, globular fields only emerged from a sufficiently high degree of overcompleteness onwards [32, 33] or with specific prior settings and overcompleteness [27]; for non-invariant occlusive models [5, 30] globular fields always emerge alongside Gabors for any overcompleteness. The results reported here can be taken as confirming this observation for position invariant encoding. The invariant non-linear model assigns high degrees of occurrences (high ?h) to Gabor-like and to globular fields (first rows in Fig. 3f). Components with more complex structures are assigned lower occurrence frequencies. In total the model assumes a fraction between 10 and 20% of all data components to be globular. Such high percentages may be related to the high percentages of globular fields (?16-23%) measured in vivo ([29] and [5] for references). In contrast, the highest degrees of occurrences, e.g., for convolutional matching pursuit [31] seems to be assigned exclusively to Gabor features. Globular fields only emerge (alongside other non-Gabor

fields) for higher degrees of overcompleteness. A direct comparison in terms of occurrence frequencies is difficult because the linear models to not infer occurrence frequencies from data. The closest match to such frequencies would be an (inverse) sparsity which is set by hand for almost all linear approaches. The reason is the use of MAP-based point-estimates while our approach uses a more probabilistic posterior estimate. 7

Because of their separate encoding of features and positions, all models with separate position encoding can represent high degrees of over-completeness. Convolutional matching pursuit [31] shows results for up to 64 filters of size 8 ? 8. With 8 horizontal and 8 vertical shifts, the number of noninvariant components would amount to 8 ? 8 ? 64 = 3136. Convolutional sparse coding [19] reports results by assuming 128 components for 9 ? 9 patches.The number of non-invariant components would therefore amount to 10, 368. For our network we obtained results for up to 100 components of size 16 ? 16. With 16 horizontal and 16 vertical shift this amounts to 25, 600 noninvariant components. In terms of components per observed variable, invariant models are therefore now computationally feasible in a regime the visual cortex is estimated to operate in [33]. The hidden units associated with component feature are fully translation invariant. In terms of neural encoding, their insensitivity to stimulus shifts would therefore place them into the category of V1 complex cells. Also globular fields or fields that seem sensitive to structures such as corners would warrant such units the label ?complex cell?. No hidden variable in the model can directly be associated with simple cell responses. However, a possible neural network implementation of the model is an explicit representation of component features at different positions. The weight sharing of the model would be lost but units with explicit non-invariant representation could correspond to simple cells. While such a correspondence can connect our predictions to experimental studies of simple cells, recently developed approaches for the estimation of translation invariant cell responses [34, 35] can represent a more direct connection. To approximately implement the non-linear generative model neurally, the integration of information would have to be a very active process. In contrast to passive pooling mechanisms across units representing linear filters (such as simple cells), it would involve neural units with explicit position encoding. Such units would control or ?gate? the information transfer from simple cells to downstream complex cells. As such our probabilistic model can be related to ideas of active control units for individual components [6, 7, 10, 11, 36] (also compare [37]). A notable difference to all these models is that the here studied approach allows to interpret active control as optimal inference w.r.t. a generative model of translations and occlusions. Future work can go in different directions. Different transformations could be considered or learned [37], explicit modeling in time could be incorporated (compare [17]), and/or further hierarchical stages could be considered. The crucial challenge all such developments face are computational intractabilities due to large combinatorial hidden spaces. Base on the presented results, we believe, however, that advances in analytical and computational training technology will enable an increasingly sophisticated modeling of image patches in the future. Acknowledgement. We

thank Richard E. Turner for helpful discussions and acknowledge funding by DFG grant LU 1196/4-2.

2 References

[1] D. Mumford and B. Gidas. Stochastic models for generic images. Q. Appl. Math., 59:85?111, 2001. [2] J. L?ucke, R. Turner, M. Sahani, and M. Henniges. Occlusive Components Analysis. NIPS, 22:1069?77, 2009. [3] Nicolas LeRoux, Nicolas Heess, Jamie Shotton, and John Winn. Learning a generative model of images by factoring appearance and shape. Neural Computation, 23:593?650, 2011. [4] D. Zoran and Y. Weiss. Natural images, Gaussian mixtures and dead leaves. NIPS, 25:1745?1753, 2012. [5] J. Bornschein, M. Henniges, and J. L?ucke. Are V1 receptive fields shaped by low-level visual occlusions? A comparative study. PLoS Computational Biology, 9(6):e1003062. [6] G. E. Hinton. A parallel computation that assigns canonical object-based frames of reference. In Proc. IJCAI, pages 683?685, 1981. [7] C. H. Anderson and D. C. Van Essen. Shifter circuits: a computational strategy for dynamic aspects of visual processing. PNAS, 84(17):6297?6301, 1987. [8] M. Lades, J. Vorbr?uggen, J. Buhmann, J. Lange, C. v. d. Malsburg, R. W?urtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. IEEE Transactions on Computers, 42(3):300?311, 1993. [9] A. Hyv?arinen and P. Hoyer. Emergence of phase- and shift-invariant features by decomposition of natural images into independent feature subspaces. Neural Computation, 12(7):1705?20, 2000. [10] D. W. Arathorn. Map-Seeking circuits in Visual Cognition ? A Computational Mechanism for Biological and Machine Vision. Standford Univ. Press, Stanford, California, 2002.

8

[11] J. L?ucke, C. Keck, and C. von der Malsburg. Rapid convergence to feature layer correspondences. Neural Computation, 20(10):2441?2463, 2008. [12] Y. LeCun, K. Kavukcuoglu, and C. Farabet. Convolutional networks and applications in vision. Proceedings of 2010 IEEE International Symposium on Circuits and Systems, pages 253?6, 2010. [13] Y. Hu, K. Zhai, S. Williamson, and J. Boyd-Graber. Modeling Images using Transformed Indian Buffet Processes. In ICML, 2012. [14] N. Jojic and B. Frey. Learning flexible sprites in video layers. In CVPR, 2001. [15] C. K. I. Williams and M. K. Titsias. Greedy learning of multiple objects in images using robust statistics and factorial learning. Neural Computation, 16:1039?62, 2004. [16] J. B. Tenenbaum and W. T. Freeman. Separating Style and Content with Bilinear Models. Neural Computation, 12(6):1247?83, 2000. [17] P. Berkes, R. E. Turner, and M. Sahani. A structured model of video reproduces primary visual cortical organisation. PLoS Computational Biology, 5(9):e1000495, 2009. [18] C. F. Cadieu and B. A. Olshausen. Learning intermediate-level representations of form and motion from natural movies. Neural Computation, 24(4):827?866, 2012. [19] K. Kavukcuoglu, P. Sermanet, Y.L. Boureau, K. Gregor, M. Mathieu, and Y. LeCun. Learning convolutional feature hierarchies for visual recognition. NIPS,

23:14, 2010. [20] K. Gregor and Y. LeCun. Efficient learning of sparse invariant representations. CoRR, abs/1105.5307, 2011. [21] J. Eggert, H. Wersing, and E. K?orner. Transformation-invariant representation and NMF. In 2004 IEEE International Joint Conference on Neural Networks, pages 2535?39, 2004. [22] Z. Dai and J. L?ucke. Unsupervised learning of translation invariant occlusive components. In CVPR, pages 2400?2407. 2012. [23] J. L?ucke and J. Eggert. Expectation truncation and the benefits of preselection in training generative models. Journal of Machine Learning Research, 11:2855?900, 2010. [24] J. H. van Hateren and A. van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. Proceedings of the Royal Society of London B, 265:359?66, 1998. [25] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. Nature Neuroscience, 211(11):1019 ? 1025, 1999. [26] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, volume 25, pages 1106?1114, 2012. [27] M. Rehn and F. T. Sommer. A network that uses few active neurones to code visual input predicts the diverse shapes of cortical receptive fields. Journal of Computational Neuroscience, 22(2):135?46, 2007. [28] J. L?ucke. Receptive field self-organization in a model of the fine-structure in V1 cortical columns. Neural Computation, 21(10):2805?45, 2009. [29] D. L. Ringach. Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. Journal of Neurophysiology, 88:455?63, 2002. [30] G. Puertas, J. Bornschein, and J. L?ucke. The maximal causes of natural scenes are edge filters. In NIPS, volume 23, pages 1939?1947. 2010. [31] A. Szlam, K. Kavukcuoglu, and Y. LeCun. Convolutional matching pursuit and dictionary training. arXiv preprint arXiv:1010.0422, 2010. [32] B. A. Olshausen, C. F. Cadieu, and D. K. Warland. Learning real and complex overcomplete representations from the statistics of natural images. volume 7446, page 74460S. SPIE, 2009. [33] B. A. Olshausen. Highly overcomplete sparse coding. In Proc. of HVEI, page 86510S, 2013. [34] M. Eickenberg, R.J. Rowekamp, M. Kouh, and T.O. Sharpee. Characterizing responses of translationinvariant neurons to natural stimuli: maximally informative invariant dimensions. Neural Computation, 24(9):2384?421, 2012. [35] B. Vintch, A. Zaharia, J.A. Movshon, and E.P. Simoncelli. Efficient and direct estimation of a neural subunit model for sensory coding. In Proc. of NIPS, pages 3113?3121, 2012. [36] B. Olshausen, C. Anderson, and D. Van Essen. A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. J Neuroscience, 13(11):4700?4719, 1993. [37] R. Memisevic and G. E. Hinton. Learning to represent spatial transformations with factored higher-order Boltzmann machines. Neural Computation, 22(6):1473?1492, 2010. [38] M.J.D. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. The Computer Journal, 7(2):155?162, 1964.

9