COMPACT PART-BASED IMAGE REPRESENTATIONS

Marc Goessling  
Department of Statistics  
University of Chicago  
Chicago, IL 60637, USA  
goessling@galton.uchicago.edu

Yali Amit  
Departments of Statistics and Computer Science  
University of Chicago  
Chicago, IL 60637, USA  
amit@galton.uchicago.edu

ABSTRACT

Learning compact, interpretable image representations is a very natural task which has not been solved satisfactorily even for simple classes of binary images. In this paper, we review various ways of composing parts (or experts) for binary data and argue that competitive forms of interaction are best suited to learn low-dimensional representations. We propose a new rule which discourages parts from learning similar structures and which penalizes opposing expert opinions strongly so that abstaining from voting becomes more attractive. Using a process of oversimplification and correction we show in experiments that very intuitive models can be obtained.

1 INTRODUCTION

In recent years, multi-layer image representations in the form of deep networks have drastically improved the performance for many classification tasks. These models use a cascade of learned nonlinear functions (e.g. Lee et al., 2009a) or build upon transformations which yield desired invariances and stability properties (Bruna & Mallat, 2013). While being effective for image classification the corresponding representations are typically high-dimensional and individual features can be unintuitive. Even for simple image classes, the basic task of learning a compact and interpretable representation has not yet been solved in a satisfactory manner. For example, the most natural representation of the letter T is in terms of a vertical and a horizontal bar. Consequently, this class can efficiently be represented by six coordinates, corresponding to the row/column location and the orientation of the bars. In this work, we will learn such a representation from as little as 10 examples. Apart from being intuitively appealing, low-dimensional part-based representations can be useful in order to obtain coarse descriptions for scene analysis tasks.

Part-based image models, also known as compositional models (Jin & Geman, 2006), use object parts as basic building blocks which traditionally are derived from a collection of image patches. The used patches can be chosen from random locations with random sizes (Borenstein & Ullman, 2002), from a fixed partition of the image grid (Pal et al., 2002), from a regular grid of overlapping windows with a fixed size (Amit & Trouvé, 2007), or based on interest point detectors (Leibe et al., 2008; Ommer & Buhmann, 2010). The parts are guaranteed to have a local descriptive scope because they are explicitly restricted to certain subregions. After the parts are learned, a next step is often to model their spatial relationship. However, due to the hard and rather arbitrary part support choices, samples from part-based generative models typically show some artifacts along the part boundaries. Splitting stable structures and subsequently studying how to recombine the parts complicates the learning process.

A different approach are deep belief networks (Lee et al., 2009a) which consist of an extensive number of experts, typically learned from large datasets. Compared to compositional models, deep
networks usually incorporate less prior knowledge and are hence more flexible. In particular, the ‘area of responsibility’ for the experts is not determined a-priori. Samples from deep generative models usually look very realistic and do not show artifacts as in part-based models. On the other hand, individual experts are not always semantically meaningful and the described structures are sometimes more global than desired. Since the representation is less explicit pose estimation tasks, where the constituting parts of articulated objects have to be located, are not as straightforward as with part-based models. Moreover, there is often strong dependency among the experts and as Bengio et al. (2013) point out learning to disentangle the factors of variation remains a key challenge in deep networks.

The aim of this work is to learn a robust representation by discovering a parsimonious set of parts corresponding to the largest stable structures in the data. We emphasize that the focus is not on achieving state-of-the-art performance in terms of classification rates or likelihoods but rather on learning simple models from few examples. The parts are learned from whole images rather than image patches and are not restricted in their support. It is therefore of key importance to define interaction rules which cause competition over the image support. We start in section 2 by presenting various composition rules and discussing their impact on the resulting representation. A new rule is then introduced which is particularly well suited to learn compact representations. In section 3 we describe an EM-style learning procedure for models with extremal composition rules, providing a batch as well as an online version. Moreover, we propose a sequential initialization method which can be described as a process of oversimplification and correction. Results for a synthetic dataset and handwritten letters are presented in section 4.

2 COMPOSITION RULES

The (binary) image data $I$ is modeled through a Bernoulli distribution $P(I | \mu)$, i.e., pixels are conditionally independent. The global template $\mu$ is a composition of part templates $\mu_1, \ldots, \mu_K$ which are defined on the entire image grid. Composition rules (also known as patchwork operation, mixing function or voting scheme) define how exactly the parts are combined in order to create the composed template. These rules are the generative counterpart of activation functions in feed-forward neural networks. Formally, a composition rule is a function $\gamma : [0, 1]^K \rightarrow [0, 1]$ with a varying number $K$ of arguments. The composed model $\mu(x)$ is obtained by applying the composition rule at each pixel

$$\mu(x) = \gamma(\mu_1(x), \ldots, \mu_K(x)).$$

Of special interest is the ability of a part to have no opinion. By that we mean there exists a value $\eta \in [0, 1]$ such that

$$\gamma(\eta, p_1, \ldots, p_K) = \gamma(p_1, \ldots, p_K)$$

for all $p_1, \ldots, p_K \in [0, 1]$ (and all $K$). In other words, if a part takes on the value $\eta$ it refrains from voting in that region.

In the following we consider two classes of models. The first class is more natural for binary images and is referred to as write-black models (Saund, 1995). In these models the default variable state is ‘off’ (white pixel). Underlying factors are now able to turn variables ‘on’ (black pixel). If multiple causes for a variable are present the state will still be ‘on’. For example, in figure-ground segmentations (white corresponding to background and black corresponding to foreground) this type of model seems appropriate. Composition rules for write-black models encode ‘no opinion’ through $\mu_k(x) = 0$ and we refer to them as asymmetric rules. Note that the value 0 is used for image regions far away from the part support. A template probability $\mu_k(x) = \frac{1}{2}$ on the other hand is used for pixels which are close to the boundary of the part support and hence rather corresponds to ‘not sure’. In these models, samples from individual part templates will look like actual object parts.

The second class we consider are write-white-and-black models (Saund, 1995). In such models parts are able to cast votes in favor of ‘on’ as well as ‘off’. Composition rules for write-white-and-black models encode ‘no opinion’ through $\mu_k(x) = \frac{1}{2}$, we refer to them as symmetric rules. Note that ‘not sure’ is also encoded through $\mu_k(x) = \frac{1}{4}$, so this value can have either of the two meanings. Samples from single parts will not look like actual object parts because about half of the pixels in the background region will be turned on.
2.1 ASYMMETRIC RULES

**Noisy OR**  A straightforward composition rule for write-black models is the disjunctive composition

\[ \gamma(p_1, \ldots, p_K) = 1 - \prod_k (1 - p_k). \]

The composed probability is just the probability of observing at least one success when drawing independently from Bernoulli distributions with probabilities \( p_1, \ldots, p_K \). This rule was used in (Saund, 1995).

**Sum of odds**  It is argued in (Dayan & Zemel, 1995) that the noisy-or rule offers little incentive for the parts to focus on different structures in the data. A more competitive composition rule is proposed which is of the form

\[ \gamma(p_1, \ldots, p_K) = 1 - \frac{1}{1 + \sum_k p_k/(1 - p_k)}. \]

This rule can be motivated as the probability of observing a success when drawing independently from Bernoulli distributions with probabilities \( p_k \) conditioned on observing at most one success. It is easy to see that the composed odds are just the sum of the individual odds. While in a global mixture model exactly one component is responsible for generating the whole observation, here exactly one part is responsible for each dimension that is turned ‘on’. So, in contrast to the noisy-or rule the responsibility for a single pixel is not shared.

**Maximum**  The most extreme form of competition is achieved through the max rule

\[ \gamma(p_1, \ldots, p_K) = \max_k p_k. \]

Such a rule was used in (Lücke & Sahani, 2008). Since only the strongest template votes, parts have no incentive to represent structures which are already present unless their opinion is the most extreme one. Consequently, parts tend to focus on different aspects of the data. In contrast to the noisy-or and the sum-of-odds rule, likelihoods cannot be improved by using the same part multiple times. Other than for the sum-of-odds rule, with this composition rule for each pixel it is even known which part is responsible. This fact allows to use an analytic formula in the M-step of the EM learning procedures for such models.

Figure 1 shows a plot of the different asymmetric composition rules. We see that the max rule is flat at 0 and hence inhibits the parts from leaving the no-opinion state.

2.2 SYMMETRIC RULES

**Average**  An intuitive composition rule for write-white-and-black models is the simple average

\[ \gamma(p_1, \ldots, p_K) = \frac{1}{K} \sum_{k=1}^K p_k. \]

The composed template can be interpreted as an equal mixture of the individual part templates. This rule was used in (Amit & Trouvè, 2007). Note however that with this composition it is impossible to have no opinion. Consequently, the support of the parts has to be restricted manually.

**Sum of log-odds**  Another popular composition rule is the sum of log-odds

\[ \gamma(p_1, \ldots, p_K) = \sigma \left( \sum_k \log \frac{p_k(x)}{1 - p_k(x)} \right) \]

where \( \sigma(t) = 1/(1 + e^{-t}) \) is the logistic function (the inverse of the logit function). This type of composition is used in restricted Boltzmann machines (Hinton, 2002) and sigmoid belief networks (Neal, 1992). A sum followed by the logistic link function is also used in generalized linear models for binary data. This includes logistic PCA (Schein et al., 2003), latent trait models (Bartholomew...
et al., 2011), exponential family sparse coding (Lee et al., 2009b) and binary matrix factorization (Meeds et al., 2006). ‘No opinion’ is expressed through log-odds of 0, i.e., probabilities of $1/2$. Since the first step is to sum up the individual votes (casted in terms of log-odds), opinions of experts voting in the opposite direction can be completely canceled out. Indeed, as seen in figure 1, even if $p_1 = 0.7$ the composed probability can be arbitrarily close to 0, just by adopting an extreme opinion in $p_2$. As a result, there is little pressure to adopt no opinion. At the same time, two very similar experts may complement each other without diminishing the likelihood compared to a single strong expert.

Normalized sum A more competitive composition rule is proposed in (Saund, 1995), which for active disagreement results in a net uncertainty. For $p_k \in \{0, \frac{1}{2}, 1\}$ the rule is

$$\gamma(p_1, \ldots, p_K) = \frac{1}{2} \left( \frac{\sum_k (p_k - \frac{1}{2})}{\sum_k |p_k - \frac{1}{2}|} + 1 \right)$$

with linear interpolation in between. However, due to the computational cost for linear interpolation a more tractable approximation is actually used for their experiments.

Maximum minus minimum We propose a new rule which reduces redundancy among parts and incentivizes vote abstention at the same time. Analogously to the max composition, we would like to rule out the possibility to increase the likelihood just by placing down the same part multiple times. This is achieved by using only the most extreme opinion. On the other hand, similarly to the normalized sum, opposing opinions should result in a limitation of the maximum achievable likelihood. This naturally leads to the max-minus-min rule

$$\gamma(p_1, \ldots, p_K) = q + (\max_k p_k - q)_+ - (\min_k p_k - q)_-$$

where the subscript $+/-$ denotes the positive/negative part of a real number (and $q = \frac{1}{2}$). To our knowledge, such a composition has not been used before.

A comparison of the symmetric composition rules is shown in figure 1. To further illustrate the difference between a summation-based composition (like the sum of log-odds) and the extremal
Figure 2: Log-likelihood functions for a two-part model with sum-of-logodds composition (left) and max-minus-min composition (right).

max-minus-min rule we compute the corresponding log-likelihood functions in a simple example. Consider the ground-truth model which creates completely white images with probability $\frac{1}{4}$, completely black images with probability $\frac{1}{4}$ and random images (each pixel is drawn independently from a Bernoulli-$\frac{1}{2}$ distribution) with probability $\frac{1}{2}$. We attempt to learn this model by training two parts which can be combined using the sum-of-logodds rule or the max-minus-min rule, respectively. For simplicity we reduce the part templates to a single parameter $\mu_k(x) = p_k$ ($k = 1, 2$), i.e., each pixel has the same chance of being turned on. We show the resulting (expected) log-likelihood functions in figure 2 (for the case of a large image resolution). The global maximum is attained at $(1, 0)$ and $(0, 1)$. As we see, if the initial parameters are on the same side of 0.5 (top-right or bottom-left quadrant) gradient descent in the summation-based model will change both parameters in the same direction (i.e, will increase both values or will decrease both values). This may partially explain why restricted Boltzmann machines (Hinton, 2002) with randomly initialized weights have a tendency to yield multiple similar experts. The situation is different for the max-minus-min composition. If both parameters are on the same side of 0.5 then moving along the gradient direction would only change the more extreme value while the other part would remain close to the no-opinion state (and would hence be ‘available’ in later stages of the learning process). Note however that we are not using gradient descent to train the max-minus-min model but rather employ the EM algorithm (see next section). The plot also suggests that it could be beneficial to use a sequential initialization procedure where additional parts take care of structures which cannot be explained by the existing ones.

3 Inference and Learning

We use an (approximate) EM procedure to learn max-minus-min models. If part transformations like shifts and rotations are modeled explicitly then each part template $\mu_k$ provides multiple transformed versions $\mu_{k,t} = \Phi_t(\mu_k)$ where $t$ denotes the transformation. This allows to share parameters among all transformed version. In the E-step, for a given image $I$ and current parts $\mu_k$ we have to find the part configuration which is most probable to have generated the data. We propose to use a sequential inference procedure similar to matching pursuit (Mallat & Zhang, 1993). One starts with the part which best explains the image data and keeps adding the part that most increases the likelihood, until no improvement is possible anymore. Further details about this procedure are given in the supplementary material. This yields for each image a representation $(\mu_{1,t_1}, \ldots, \mu_{K,t_K})$. We define

$$k^*(x) = \arg \max_k \mu_{k,t_k}(x), \quad \ell^*(x) = \arg \min_k \mu_{k,t_k}(x)$$

provided that the maximum is larger than $q$ and the minimum is smaller than $q$, respectively (otherwise we leave $k^*(x)$ or $\ell^*(x)$ undefined). In words, $k^*(x)$ and $\ell^*(x)$ are the parts with the most
extreme opinions for pixel \( x \). In the M-step we update the parts by computing

\[
\mu_k(x) = \frac{\sum_n \mathbb{1}\{k = k_n^*(x) \text{ or } k = \ell_n^*(x)\} \Phi_{t_{nk}}^{-1}(I_n(x))}{\sum_n \mathbb{1}\{k = k_n^*(x) \text{ or } k = \ell_n^*(x)\}}
\]

where \( k_n^* \), \( \ell_n^* \) comprise the maximizers respectively minimizers for the \( n \)-th training example \( I_n \) and \( t_{nk} \) is the transformation corresponding to \( k \)-th part in the representation of \( I_n \). For regularization purposes we add pseudocounts of 1 to the expression (which can be interpreted as a uniform prior on the pixel probabilities). It should be emphasized that the analytic update formula is a distinct feature of extremal composition rules. With other composition rules the responsibility for single pixels is split among many parts and consequently gradient descent methods are required to perform the M-step in those models. Also note that for \( q = 0 \) the max-minus-min model reduces to the max model. In that special case the M-step formula only involves \( k^* \) but not \( \ell^* \).

3.1 ONLINE LEARNING AND SEQUENTIAL INITIALIZATION

The online update rules are straightforward and can be found in the supplementary material. Since the learning problem is non-convex, it is crucial to have a good initialization. In accordance with our attempt to learn a parsimonious representation we start off with a single global template derived from the first training image and add more parts later on. The idea is to use ‘oversimplified’ models in the sense that they try to explain new examples through shifted and rotated versions of the parts learned so far. These models are then ‘corrected’ by appending residual images (i.e., the difference between a training example and the best model explanation) as additional parts. This approach is in stark contrast to the typical bottom-up grouping of local structures in part-based composition models. For example, (Fidler & Leonardis, 2007; Zhu et al., 2008) start with elementary edge features and combine simple structures into more complex ones through a hierarchical clustering procedure.

3.2 GEOMETRIC COMPONENT

The spatial arrangement of the parts can be modeled as a joint Gaussian distribution on locations and orientations. This is easy to learn and requires only to compute the mean and covariance of the training configurations provided by the inference procedure. As emphasized in (Bruna & Mallat, 2013), a very desirable property of a representation is that intraclass deformations are linearized. As the experiments in the next section confirm, our part-based representation transforms the complex deformation orbit (in the original space) into a linear space in which a Gaussian distribution satisfactorily describes the deformations.

4 EXPERIMENTS

4.1 HANDWRITTEN LETTERS

We use letters from the TiCC handwritten characters dataset (van der Maaten, 2009) to test the max composition model. Each sample comes with a label for the writer of the corresponding character. Figure 3 visualizes the online learning process for the letter T, using the first sample of the first 10 writers. The first part is initialized by the first training case. The second part is initialized through the residual image of the second example. Every additional image then updates both parts. After 10 examples the learning process has converged to a vertical and a horizontal bar. Samples from the learned spatial distribution look realistic and cover the principal deformations of the class. We also learned up to 4 parts for the other letters\(^1\), using the first sample of the first 20 writers. The results are shown in figure 4. Most parts correspond to natural elements of the character class. Note that in contrast to (Lake et al., 2013) no motion information was necessary to learn these character primitives.

4.2 A SYNTHETIC WRITE-WHITE-AND-BLACK MODEL

We compare the max-minus-min model to denoising autoencoders (Vincent et al., 2008) and restricted Boltzmann machines (Hinton, 2002) on a synthetic model for binary images of size 6 × 6

\(^1\)The letter X is missing from the dataset.
Figure 3: Learning a part model for the letter T. **First row:** 10 samples of the letter T. **Second and third row:** Online learning of two parts. Shown are the two templates at step $i = 1, \ldots, 10$ (blue corresponding to 0, yellow corresponding to 1). **Fourth row:** Sampled part configurations using a multivariate Gaussian distribution on the spatial arrangement.

Figure 4: Parts learned from 20 examples per class. Each part is plotted at its mean location with mean orientation. For each pixel the color (red, green, blue, magenta) indicates the maximum part and the intensity visualizes the template value (white corresponding to 0, color corresponding to 1).

pixels. The image grid is divided into a regular grid of four quadrants. Each quadrant is independently activated with probability $\frac{1}{2}$. An activated quadrant is either entirely black or entirely white, each with probability $\frac{1}{2}$. For a non-activated quadrant the pixels are drawn independently from a Bernoulli-$\frac{1}{2}$ distribution. Samples from this model can be found in the supplementary material. The task is to recover the 8 underlying parts corresponding to the four quadrants with two polarities. The learning algorithm for the max-minus-min model typically converged after a few iterations. In figure 5 we visualize the obtained templates after 1, 2 and 5 iterations when using 100 training examples. As we see, the extremal model is able to almost perfectly recover the original parts. The denoising autoencoder and the restricted Boltzmann machine were run for 100 epochs and the learning rates were tuned. For the denoising autoencoder we also tuned the corruption level. Figure 5 shows a comparison with the max-minus-min model in terms of the cross-entropy (or negative log-likelihood of the composed templates) on a test set of 1000 samples. The reconstruction error for the max-minus-min model is much lower. This is because the sum-based models mix up multiple ground-truth parts (we visualize this in the supplementary material). For a fair comparison we left out the sequential initialization procedure (as well as the transformation modeling) and initialized all models with completely random templates. So, the performance gain is indeed due to the more
competitive interaction scheme. In the supplementary material we also contrast our representation with dictionary learning for sparse coding.

5 CONCLUSION AND FUTURE WORK

We studied composition rules for part-based models and discussed their adequacy for learning compact representations. Inference and learning procedures for models with extremal composition rules were then proposed and their performance was tested in experiments. An alternative to competitive interaction rules would be to use a prior on the part parameters (e.g., an $L_1$-penalty on the log-odds). However, this would create a bias which affects all parts while our approach allows to use maximum-likelihood updates.

The focus here was on binary image data. A natural next step is to study part interactions in real-valued images. This includes considering composition rules for variances (in addition to means) and achieving continuity of intensity values for transitions from one part into another. A further opportunity for future work is the extension of the model to multiple layers. This could be done through a hierarchy of Gaussian distributions for spatial part arrangements or for more general data via multiple max-min layers.

REFERENCES

Amit, Yali and Trouv´e, Alain. Pop: Patchwork of parts models for object recognition. International Journal of Computer Vision, 75(2):267–282, 2007.

Bartholomew, David J., Knott, Martin, and Moustaki, Irini. Latent variable models and factor analysis. Wiley, 2011.

Bengio, Yoshua, Courville, Aaron, and Vincent, Pascal. Representation learning: A review and new perspectives. Pattern Analysis and Machine Intelligence, 35(8):1798–1828, 2013.

Borenstein, Eran and Ullman, Shimon. Class-specific, top-down segmentation. In Computer Vision–ECCV, pp. 109–122. 2002.

Bruna, Joan and Mallat, St´ephane. Invariant scattering convolution networks. Pattern Analysis and Machine Intelligence, 35(8):1872–1886, 2013.

Dayan, Peter and Zemel, Richard S. Competition and multiple cause models. Neural Computation, 7(3):565–579, 1995.

Fidler, Sanja and Leonardis, Ales. Towards scalable representations of object categories: Learning a hierarchy of parts. In Computer Vision and Pattern Recognition, pp. 1–8, 2007.
Hinton, Geoffrey E. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

Jin, Ya and Geman, Stuart. Context and hierarchy in a probabilistic image model. In *Computer Vision and Pattern Recognition*, volume 2, pp. 2145–2152, 2006.

Lake, Brenden M., Salakhutdinov, Ruslan, and Tenenbaum, Josh. One-shot learning by inverting a compositional causal process. In *Advances in Neural Information Processing Systems*, pp. 2526–2534, 2013.

Lee, Honglak, Grosse, Roger, Ranganath, Rajesh, and Ng, Andrew Y. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *International Conference on Machine Learning*, pp. 609–616, 2009a.

Lee, Honglak, Raina, Rajat, Teichman, Alex, and Ng, Andrew Y. Exponential family sparse coding with application to self-taught learning. In *IJCAI*, volume 9, pp. 1113–1119, 2009b.

Leibe, Bastian, Leonardis, Aleš, and Schiele, Bernt. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1-3):259–289, 2008.

Lücke, Jörg and Sahani, Maneesh. Maximal causes for non-linear component extraction. *The Journal of Machine Learning Research*, 9:1227–1267, 2008.

Mallat, Stéphane G. and Zhang, Zhifeng. Matching pursuits with time-frequency dictionaries. *Signal Processing*, 41(12):3397–3415, 1993.

Meeds, Edward, Ghahramani, Zoubin, Neal, Radford M., and Roweis, Sam T. Modeling dyadic data with binary latent factors. In *Advances in Neural Information Processing Systems*, pp. 977–984, 2006.

Neal, Radford M. Connectionist learning of belief networks. *Artificial intelligence*, 56(1):71–113, 1992.

Ommer, Björn and Buhmann, Joachim M. Learning the compositional nature of visual object categories for recognition. *Pattern Analysis and Machine Intelligence*, 32(3):501–516, 2010.

Pal, Chris, Frey, Brendan J., and Jojic, Nebojsa. Learning montages of transformed latent images as representations of objects that change in appearance. In *Computer Vision–ECCV*, pp. 715–731. 2002.

Saund, Eric. A multiple cause mixture model for unsupervised learning. *Neural Computation*, 7(1):51–71, 1995.

Schein, Andrew I., Saul, Lawrence K., and Ungar, Lyle H. A generalized linear model for principal component analysis of binary data. In *International Workshop on Artificial Intelligence and Statistics*, volume 38, pp. 46, 2003.

van der Maaten, Laurens. A new benchmark dataset for handwritten character recognition. Technical report, TiCC TR 2009-002, Tilburg University, 2009.

Vincent, Pascal, Larochelle, Hugo, Bengio, Yoshua, and Manzagol, Pierre-Antoine. Extracting and composing robust features with denoising autoencoders. In *International Conference on Machine Learning*, pp. 1096–1103, 2008.

Zhu, Long L., Lin, Chenxi, Huang, Haoda, Chen, Yuanhao, and Yuille, Alan. Unsupervised structure learning: Hierarchical recursive composition, suspicious coincidence and competitive exclusion. In *Computer Vision–ECCV*, pp. 759–773. 2008.
Supplementary Material – Compact Part-Based Image Representations

Marc Goessling  
Department of Statistics  
University of Chicago  
Chicago, IL 60637, USA  
goessling@galton.uchicago.edu

Yali Amit  
Department of Statistics  
University of Chicago  
Chicago, IL 60637, USA  
amit@galton.uchicago.edu

1 Sequential Inference Procedure

Given an image \(I\) and parts \(\mu_k\) the inference task is to determine the posterior distribution on part configurations given the observation, or at least finding the part configuration which is most likely to have generated the data. For our purposes it is important that the inference procedure yields compact image descriptions. Since the templates will describe rather large structures, we assume that only one of the transformed versions is present in each image. We mention a few different approaches before presenting the sequential procedure which we use in our experiments. In restricted Boltzmann machines (Hinton, 2002) parts are evaluated independently which is computationally efficient but will activate all parts that match the data sufficiently well, rather than providing the sparsest possible activation which can explain the data. In other works, the indicator variables for part presence are relaxed to real values in \([0, 1]\) which makes it difficult to deal with additional constraints on the configurations. For example, Dayan & Zemel (1995) and Vincent et al. (2008) use simple mean-field approximations. In (Saund, 1995) gradient descent is used starting from a point with all coordinates equal to \(\frac{1}{2}\). This iterative procedure becomes inefficient if the number of possible parts is much larger than the typical number of active parts. Lücke & Sahani (2008) use a truncated search which evaluates all part configurations with a small number of active components.

We propose a simple sequential inference procedure. One starts with the part which best explains the image data and keeps adding the part that most increases the likelihood. The procedure ends when the likelihood cannot be improved anymore. This yields a sparse activation in a single sequential pass because parts are only added if they are able to explain structures which have not been explained before. Note that this procedure is quite similar to matching pursuit (Mallat & Zhang, 1993) for sparse coding, just maximizing the Bernoulli likelihood instead of minimizing the squared error. This sequential fitting works well for write-white-and-black models. However, since write-black models code ‘no opinion’ through a probability of 0 (rather than \(\frac{1}{2}\)) the value of \(P(I | \mu_k)\) will

Figure 1: First panel: Scene to be analyzed. Second panel: Noisy version. Third panel: Resolved scene using robustified templates (the digits are detected in the order red, green, blue, yellow, magenta). Fourth panel: First detected digit using the original templates.
heavily depend on which structures other than the one described by \( \mu_k \) are present in the image \( I \). Consequently, it is easy to run into situations where

\[
P(I \mid \mu_0) \ll P(I \mid \mu_0') \quad \text{but} \quad P(I \mid \gamma(\mu_0, \mu_1, \ldots, \mu_K)) \gg P(I \mid \gamma(\mu_0', \mu_1, \ldots, \mu_K)),
\]

i.e., the choice of the first part may lead to a poor local maximum. The problem is that the first part tries to explain the entire image. As a simple fix we propose to use truncated templates

\[
\tilde{\mu}_k(x) = \max \left( \frac{1}{2}, \mu_k(x) \right)
\]

instead. This eliminates the impact of the data in the background region of the part. Indeed, \( P(I \mid \tilde{\mu}_k) \) only depends on the image data \( I(x) \) for those \( x \) which satisfy \( \mu_k(x) > \frac{1}{2} \), i.e., it only depends on the data in the ‘support’ of part \( k \). A similar idea to robustify templates was also used in (Williams & Titsias, 2004). There the original templates are mixed with a uniform distribution. In the case of binary data this amounts to \( \alpha \mu(x) + (1 - \alpha)/2 \) for some \( \alpha \in (0, 1) \), i.e., a convex combination between \( \mu \) and \( \frac{1}{2} \) is formed. For a write-black image model this is a less effective transformation than our truncation. We illustrate the effectiveness of the modification through a simple scene analysis example. 5 digits are placed at random locations in the image according to a write-black model. The task is to recover the identity and locations of the digits in a noisy version of this scene. Using the robustified templates the scene can be resolved perfectly, as shown in figure 1. However, with the original templates the first digit which is placed down corresponds to the largest structure in the image that can be explained by a single template, even though the fit is not very good.

2 ONLINE LEARNING

The M-step updates for the online version are

\[
\mu_k(x) \leftarrow \frac{N_k(x)\mu_k(x) + 1 \{ k = k_{n}^* (x) \text{ or } k = \ell_{n}^* (x) \} \Phi_{t_{n}}^{-1}(I_n)(x)}{N_k(x) + 1 \{ k = k_{n}^* (x) \text{ or } k = \ell_{n}^* (x) \}},
\]

\[
N_k(x) \leftarrow N_k(x) + 1 \{ k = k_{n}^* (x) \text{ or } k = \ell_{n}^* (x) \}.
\]

The variable \( N_k(x) \) counts how many training examples have been used to compute the current estimate for part \( k \) at location \( x \).

3 SYNTHETIC EXPERIMENT

In figure 2 we visualize the parts learned from 100 training examples by a denoising autoencoder (Vincent et al., 2008) and a restricted Boltzmann machine (Hinton, 2002). As we see, the templates

![Figure 2: Parts obtained from a denoising autoencoder (top), a restricted Boltzmann machine (center) and dictionary learning for sparse coding (bottom) using 100 training samples from the synthetic model.](image-url)
Figure 3: a) Sparse coding reconstruction. b) 100 samples from the synthetic model. c) Ground-truth templates for the samples in b). d) Reconstruction of the max-minus-min model.

are combinations of multiple ground-truth parts, i.e., the factors of variation have not been disentangled successfully. This is the reason for the inferior performance (in terms of cross-entropy) in the reconstruction experiments in the main paper.

As another comparison we learned a sparse coding dictionary (Mairal et al., 2009). Note that the data is than treated as real-valued and the task is to find basis vectors whose linear combinations allow for good reconstructions in a squared error sense. The basis vectors learned from 100 samples are visualized in figure 2. Again, the ground-truth parts are not recovered. However, the basis vectors look more structured than the experts learned by denoising autoencoders or restricted Boltzmann machines. In terms of $L_2$ reconstruction error this dictionary is actually even better than the ground-truth generative model. Figure 3 visualizes the reconstructions obtained via sparse coding and via the learned max-minus-min model. The sparse coding reconstruction is visually closer to the data but much more noisy than the reconstruction of the extremal model. Indeed, the max-minus-min reconstruction is almost identical to the ground-truth templates. The reason for this different behavior is that the squared error is more forgiving for small deviations compared to cross-entropy and penalizes harder than cross-entropy if the deviation is large (around 0.5).
REFERENCES
Dayan, Peter and Zemel, Richard S. Competition and multiple cause models. *Neural Computation*, 7(3):565–579, 1995.

Hinton, Geoffrey E. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

Lücke, Jörg and Sahani, Maneesh. Maximal causes for non-linear component extraction. *The Journal of Machine Learning Research*, 9:1227–1267, 2008.

Mairal, Julien, Bach, Francis, Ponce, Jean, and Sapiro, Guillermo. Online dictionary learning for sparse coding. In *International Conference on Machine Learning*, pp. 689–696, 2009.

Mallat, Stéphane G. and Zhang, Zhifeng. Matching pursuits with time-frequency dictionaries. *Signal Processing*, 41(12):3397–3415, 1993.

Saund, Eric. A multiple cause mixture model for unsupervised learning. *Neural Computation*, 7(1):51–71, 1995.

Vincent, Pascal, Larochelle, Hugo, Bengio, Yoshua, and Manzagol, Pierre-Antoine. Extracting and composing robust features with denoising autoencoders. In *International Conference on Machine Learning*, pp. 1096–1103, 2008.

Williams, Christopher K.I. and Titsias, Michalis K. Greedy learning of multiple objects in images using robust statistics and factorial learning. *Neural Computation*, 16(5):1039–1062, 2004.