Int J Comput Vis
DOI 10.1007/s11263-013-0669-1

The Shape Boltzmann Machine: A Strong Model of Object Shape

S. M. Ali Eslami · Nicolas Heess · Christopher K. I. Williams · John Winn

Received: 4 February 2013 / Accepted: 11 October 2013
© Springer Science+Business Media New York 2013

Abstract A good model of object shape is essential in applications such as segmentation, detection, inpainting and graphics. For example, when performing segmentation, local constraints on the shapes can help where object boundaries are noisy or unclear, and global constraints can resolve ambiguities where background clutter looks similar to parts of the objects. In general, the stronger the model of shape, the more performance is improved. In this paper, we use a type of deep Boltzmann machine (Salakhutdinov and Hinton, International Conference on Artificial Intelligence and Statistics, 2009) that we call a Shape Boltzmann Machine (SBM) for the task of modeling foreground/background (binary) and parts-based (categorical) shape images. We show that the SBM characterizes a *strong* model of shape, in that samples from the model look realistic and it can generalize to generate samples that differ from training examples. We find that the SBM learns distributions that are qualitatively and quantitatively better than existing models for this task.

Keywords Shape · Generative · Deep Boltzmann machine · Sampling

S. M. A. Eslami (✉) · C. K. I. Williams
School of Informatics, University of Edinburgh, Edinburgh, UK
e-mail: s.m.eslami@sms.ed.ac.uk

C. K. I. Williams
e-mail: ckiw@inf.ed.ac.uk

N. Heess
Gatsby Computational Neuroscience Unit,
University College London, London, UK
e-mail: nheess@gatsby.ucl.ac.uk

J. Winn
Microsoft Research, Cambridge, UK
e-mail: jwinn@microsoft.com

Published online: 12 November 2013

1 Introduction

Models of the shape of an object play a crucial role in many imaging algorithms, such as those for object detection and segmentation (e.g. Borenstein et al. 2004; Winn and Jojic. 2005; Alexe et al. 2010a; Eslami and Williams 2011), inpainting (e.g. Chan and Shen 2001; Bertozzi et al. 2007; Shekhovtsov et al. 2012) and graphics (e.g. Anguelov et al. 2005). In object segmentation, local constraints on the shape, such as smoothness and continuity, can help provide correct segmentations where the object boundary is noisy or lost in shadow. More global constraints, such as ensuring the correct number of parts (legs, wheels, etc.), can resolve ambiguities where background regions look similar to an object part (e.g. Jojic et al. 2009). Shape also plays an important role in generative models of images (e.g. Frey et al. 2003; Williams and Titsias 2004; Le Roux et al. 2011; Eslami and Williams 2011). In general, the better the model of object shape, the more performance will be improved in these applications.

This paper addresses the question of how to build a *strong* probabilistic model of object shapes. We define a strong model as one which meets two requirements:

1. *Realism*—samples from the model look realistic;
2. *Generalization*—the model can generate samples that differ from training examples.

The first constraint ensures that the model captures shape characteristics at all spatial scales well enough to place probability mass only on images that belong to the 'true' shape distribution. The second constraint ensures that there are no gaps in the learned distribution, i.e. that it also covers novel unseen but valid shapes.

There have been a wide variety of approaches to modeling 2D shape. The most commonly used models are

grid-structured Markov random fields (MRFs) or conditional random fields (CRFs, e.g. Boykov and Jolly 2001). In such models, the pairwise potentials connecting neighboring pixels impose very local constraints like smoothness but are unable to capture more complex properties such as convexity or curvature, nor can they account for longer-range properties. Carefully designed high-order potentials (e.g. Kohli 2007; Komodakis 2009; Rother et al. 2009; Kohli et al. 2009; Nowozin and Lampert 2009) allow particular local or longer-range shape properties to be modeled within an MRF, but these potentials fall short of capturing *all* such properties so as to make realistic-looking samples. For example, a strong shape model of horses would know that horses have legs, heads and tails, that these parts appear in certain positions consistent with a global pose, that there are never more than four legs visible in any given image, that the legs have to support the horse's body, along with many more properties that are difficult to express in words but necessary to make the shape look plausible.

Other approaches represent shape using a level set or parameterized contour. These have different strengths and weaknesses, but all share the fundamental challenge of imposing sufficient constraints to limit the model to valid shapes while allowing for the right degree of flexibility to capture all possible shapes. For example, a common approach when using a contour (or an image) is to use a mean shape in combination with some principal directions of variation, as captured by a principal components analysis (Cootes et al. 1995) or factor analysis (Cemgil et al. 2005; Eslami and Williams 2011). Such models capture the typical global shape of an object and global variations on it (such as changes in the aspect ratio of a face). However, they cannot capture multimodal distributions, and tend to be poor at learning about local variations which affect only part of the shape (e.g. the angle of a horse's front legs).

Non-parametric approaches employ what is effectively a large database of template shapes (Gavrila 2007) or shape fragments (Borenstein et al. 2004; Kumar and Torr 2005). In the former case, because no attempt is made to understand the composition of the shape, it is impossible to generalize to novel shapes not present in the database. In the latter case, the challenge lies in how to compose the shape fragments to form valid shapes. We are not aware of any method which can generate a variety of realistic looking *whole* shapes by composing fragments. Table 1 and Fig. 1 illustrate why these existing approaches do not meet the criteria for a strong shape model.

In this paper, we consider a class of models from the learning community, known as deep Boltzmann machines (DBMs, Salakhutdinov and Hinton 2009). The main contribution of this paper is to show how a strong model of binary shape can be constructed using a form of DBM with a set of carefully chosen capacity constraints, which we call the *Shape Boltzmann Machine* (SBM). The model is a generative model of object shape and can be learned directly from training data. The capacity constraints allow training on relatively small training sets as are common e.g. for segmentation datasets. Due to its *generative* formulation the SBM can be used very flexibly, not just as a shape prior in segmentation tasks but also, for instance, to synthesize novel shapes in graphics applications, or to complete partially occluded shapes. We learn SBM models from several challenging shape datasets and evaluate them on a range of shape synthesis and completion tasks. We demonstrate that, despite the relatively small sizes of the training datasets, the learned models are both able to generate *realistic* samples and to *generalize* to generate samples that differ from images in the training dataset. We provide a detailed discussion of the roles played by the different capacity constraints in making the SBM work. We finally present an extension of the SBM that also allows it to simultaneously model the shape of multiple dependent regions such as the parts of an object, which can in turn be used, for instance, as a prior in parts-based segmentation tasks.

The remainder of the paper is structured as follows. In Sect. 2 we review several families of probability distributions that have been used in the literature to model object shape.

Table 1 Comparison of a number of different shape models

	Realism		Generalization
	Globally	Locally	
Mean e.g. Jojic and Caspi (2004)	✓	–	–
Deformation field e.g. Winn and Jojic. (2005)	–	✓	✓
Factor analysis e.g. Cemgil et al. (2005)	✓	–	✓
Fragments e.g. Borenstein et al. (2004)	–	✓	✓
Grid MRFs/CRFs e.g. Rother et al. (2004)	–	✓	✓
High-order potentials e.g. Nowozin and Lampert (2009)	Limited	✓	✓
Database e.g. Gavrila (2007)	✓	✓	–
Shape Boltzmann Machine	✓	✓	✓

(a) Mean **(b)** MRF **(c)** FA

Fig. 1 Samples generated by (**a**) a mean-only model of horse shapes, (**b**) a Markov random field model, (**c**) discrete factor analysis as defined in Eqs. 18, 19

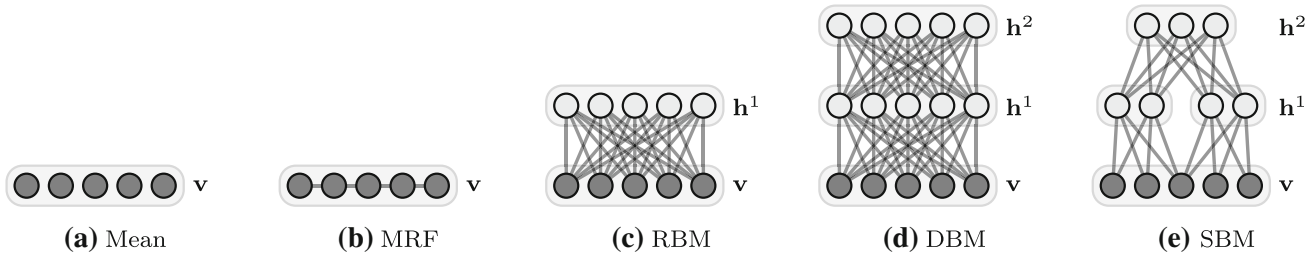

Fig. 2 Models of shape. (**a**) 1D slice of a mean model, (**b**) Markov random field in 1D, (**c**) Restricted Boltzmann machine in 1D, (**d**) Deep Boltzmann machine in 1D, (**e**) Shape Boltzmann Machine in 1D

In Sects. 3 and 4 we present the SBM and describe efficient inference and learning schemes for the model. We provide an extensive experimental evaluation in Sect. 5, and conclude with a discussion in Sects. 6 and 7.

2 Related Work

In this section we will review several undirected models suitable for modeling binary shape images. We will start with the commonly used grid-structured MRF and describe how it can be modified to form an undirected model known as the restricted Boltzmann machine (RBM). We then describe how RBMs can be stacked to form the hierarchical structure of the deep Boltzmann machine (DBM).

We will specify undirected models in terms of an energy function $E(x_1, \ldots, x_N)$ defined over the relevant set of random variables $x_1, \ldots, x_N$ (image pixels, possibly latent variables). The associated Gibbs distribution is then given by:

$$p(x_1, \ldots, x_N) = \frac{1}{Z} \exp\{-E(x_1, \ldots, x_N)\}, \tag{1}$$

where $Z = \sum_{x_1,\ldots,x_N} \exp\{-E(x_1, \ldots, x_N)\}$ is the normalization constant. We will further use v_i to denote image pixel i, and $\mathbf{v} = (v_i)^T$ to denote a column-vector of image pixels. The pixels are assumed to be binary (we consider categorical pixels in Sect. 3.2). Similarly we use h_j and $\mathbf{h} = (h_j)^T$ to refer to binary hidden variable j and a vector of hidden variables respectively.

2.1 Grid Markov Random Fields

The simplest approach is to model each shape pixel v_i independently with categorical variables whose parameters are specified by the object's mean shape (Fig. 2a). Such a 'mean model' can be expressed in terms of an energy function comprised of single-variable terms only:

$$E(\mathbf{v}; \Theta) = \sum_i f_i(v_i; b_i). \tag{2}$$

For binary images, for instance, the f_i might take the form $f_i(v_i; b_i) = -b_i v_i$, specifying the unnormalized log-probability of $v_i = 1$ which results in the normalized probability being $p(v_i = 1; b_i) = \exp(b_i)/(1 + \exp(b_i))$.

A binary grid-structured MRF defines a distribution over binary images $\mathbf{v}$ whose energy function is:

$$E(\mathbf{v}; \Theta) = \sum_i f_i(v_i; b_i) + \sum_{(i,j)} f_{ij}(v_i, v_j; w_{ij}), \tag{3}$$

where i ranges over image pixels, (i, j) ranges over grid edges between pixels i and j and the potentials are parameterized by b_i and w_{ij}, again jointly denoted by Θ. The grid structure of the MRF arises from the pairwise potentials f_{ij} shown in Fig. 2b. These potentials induce dependencies between neighboring pixels that can favor local shape properties such as connectedness or smoothness, but it is commonly accepted that grid-structured, pairwise MRFs are very limited models of global shape (e.g. Morris et al. 1996; Tjelmeland and Besag 1998).

In an attempt to capture more complex or global shape properties, much recent research has therefore focused on constructing higher-order potentials (HOPs), which take the configuration of larger groups of image pixels into account (i.e. their energy includes potentials f that depend on more than two pixel variables). The maximum number of variables per potential is referred to as the 'order' of the model. Since, in general, the cost of naïve inference (e.g. finding the most likely (MAP) configuration of the variables) in MRFs grows exponentially in the model order, there has been a strong emphasis on developing HOPs for which efficient inference schemes can be devised.

The higher order potentials in Rother et al. (2009), for instance, are defined in terms of a set of 'reference patterns' and penalize deviations of groups of pixels from these patterns. Such HOPs can be considered to be introducing an auxiliary hidden variable connected through pairwise potentials to multiple image pixels. The introduction of such hidden variables provides a powerful way to capture and learn complex properties of multiple image pixels. When such hidden variables are marginalized out they induce high-order constraints amongst the image pixels. Yet, because the model only contains pairwise potentials, both learning and inference remain tractable.

2.2 Restricted Boltzmann Machines

One model that makes heavy use of hidden variables to introduce dependencies between the observed variables is the RBM (e.g. Freund and Haussler 1994). In an RBM, a number of hidden variables $\mathbf{h}$ are used, each of which is connected to all image pixels as shown in Fig. 2c. However, unlike a grid MRF, there are no direct connections between the image pixels $\mathbf{v}$. There are also no direct connections between the hidden variables. Hence, the energy function takes the form:

$$E(\mathbf{v}, \mathbf{h}; \Theta) = \sum_i b_i v_i + \sum_{i,j} w_{ij} v_i h_j + \sum_j c_j h_j, \tag{4}$$

where i now ranges over pixels and j ranges over hidden variables. The key points to note are that the potential functions are all simple products and that the only pairwise potentials are those between each visible and each hidden variable. By learning the parameters of the potentials $\{w_{ij}, b_i, c_j\}$, the model can learn about high-order constraints in the data set.

The effect of the latent variables can be directly appreciated by considering the marginal distribution over $\mathbf{v}$ which is given by marginalizing over the hidden variables:

$$p(\mathbf{v}; \Theta) = \sum_{\mathbf{h}} \frac{1}{Z(\Theta)} \exp\{-E(\mathbf{v}, \mathbf{h}; \Theta)\}, \tag{5}$$

where the normalization constant $Z(\Theta)$ is given by $Z(\Theta) = \sum_{\mathbf{v},\mathbf{h}} \exp\{-E(\mathbf{v}, \mathbf{h}; \Theta)\}$. This marginalization allows the model to capture high-order dependencies between the visible units. In fact, the hidden units can be summed out analytically (e.g. Freund and Haussler 1994), giving rise to an alternative formulation of the RBM in terms of high-order potentials that no longer includes latent variables. The energy of this marginal distribution is given by:

$$E(\mathbf{v}; \Theta) = \sum_i f_i(v_i; b_i) + \sum_j g_j(\mathbf{v}; W_{\cdot j}), \tag{6}$$

where $f_i(v_i; b_i) = -b_i v_i$ and $g_j(\mathbf{v}) = -\log(1 + \exp\left(\sum_i w_{ij} v_i + c_j\right))$.

It is instructive to compare the form of Eq. 6 with the energy of the grid-structured MRF in Eq. 3: whereas the energy of the grid-structured MRF was comprised of unary and pair-wise terms only ($f_i(v_i)$ and $f_{ij}(v_i, v_j)$ respectively), the energy of the RBM involves unary potentials *as well as high-order* potentials, each of which is defined over all pixels $\mathbf{v}$ (the $g_j(\mathbf{v})$). There is one such high-order potential for each hidden unit, and it is these high-order potentials that allow the RBM to model considerably more complicated dependencies than, for instance, pairwise MRFs.

Whilst marginalization over the latent variables makes the high-order potentials explicit, the formulation that includes latent variables suggests an efficient inference scheme (in loose analogy to the use of latent variables for the HOPs discussed in Sect. 2.1): When written as in Eq. 4 the RBM forms a bipartite graph that has edges only between hidden and visible variables. As a consequence all hidden units are conditionally independent given the visible units—and vice versa. This property can be exploited to make inference exact and efficient. The conditional probabilities are:

$$p(v_i = 1|\mathbf{h}) = \sigma\left(\sum_j w_{ij} h_j + b_i\right), \tag{7}$$

$$p(h_j = 1|\mathbf{v}) = \sigma\left(\sum_i w_{ij} v_i + c_j\right), \tag{8}$$

where $\sigma(y) = 1/(1+\exp(-y))$ is the sigmoid function. This property allows for efficient implementations of block-Gibbs sampling where all $\mathbf{v}$ and all $\mathbf{h}$ are sampled in parallel in an alternating manner, which can be exploited during approximate learning (Hinton 2002; Tieleman 2008).

2.3 Deep Boltzmann Machines

RBMs can, in principle, approximate any binary distribution (Freund and Haussler 1994; Le Roux and Bengio 2008), but this can require an exponential number of hidden units and a similarly large amount of training data. The DBM provides a richer model by introducing additional layers of latent variables as shown in Fig. 2d. The additional layers capture high-order dependencies between the hidden variables of previous layers and so can learn about complex structure in the data using relatively few hidden units. The energy of a DBM with two layers of latent variables is given by:

$$E(\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2; \Theta) = \sum_i b_i v_i + \sum_{i,j} w^1_{ij} v_i h^1_j + \sum_j c^1_j h^1_j + \sum_{j,k} w^2_{jk} h^1_j h^2_k + \sum_k c^2_k h^2_k. \tag{9}$$

As for the RBM, the posterior distribution over the visibles is obtained by marginalization, this time with respect to both sets of hidden variables:

$$p(\mathbf{v}; \Theta) = \sum_{\mathbf{h}^1,\mathbf{h}^2} \frac{1}{Z(\Theta)} \exp\{-E(\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2; \Theta)\}, \tag{10}$$

and the normalization constant defined analogously: $Z(\Theta) = \sum_{\mathbf{v},\mathbf{h}^1,\mathbf{h}^2} \exp\{-E(\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2; \Theta)\}$.

Although exact inference is no longer possible in this model, the conditional distributions $p(\mathbf{v}|\mathbf{h}^1)$, $p(\mathbf{h}^1|\mathbf{v}, \mathbf{h}^2)$, and $p(\mathbf{h}^2|\mathbf{h}^1)$ remain factorized due to the layering:

$$p(v_i = 1|\mathbf{h}^1) = \sigma\left(\sum_j w^1_{ij} h^1_j + b_i\right), \tag{11}$$

$$p(h^1_j = 1|\mathbf{v}, \mathbf{h}^2) = \sigma\left(\sum_i w^1_{ij} v_i + \sum_k w^2_{jk} h^2_k + c^1_j\right), \tag{12}$$

$$p(h_k^2 = 1|\mathbf{h}^1) = \sigma\left(\sum_j w_{jk}^2 h_j^1 + c_k^2\right). \quad (13)$$

This allows for computationally efficient inference, either by layerwise block-Gibbs sampling from the posterior $p(\mathbf{h}^1, \mathbf{h}^2|\mathbf{v})$ (Fig. 4), or by using a mean field procedure with a fully factorized approximate posterior as described in Salakhutdinov and Hinton (2009). The layering further admits a layer-wise pre-training procedure that makes it less likely that learning will get stuck in local optima. Hence the DBM is both a rich model of binary images and a tractable one.

3 Model

RBMs and DBMs are powerful generative models, but also have many parameters. Since they are typically trained on large amounts of unlabeled data (thousands or tens of thousands of examples), this is usually less of a problem than in supervised settings. Segmented images, however, are expensive to obtain and datasets are typically small (hundreds of examples). In such a regime, RBMs and DBMs can be prone to overfitting.

In this section we will describe how we can impose a set of carefully chosen connectivity and capacity constraints on a DBM to overcome this problem: the resulting SBM formulation not only learns a model that accurately captures the properties of binary shapes, but that also generalizes well, even when trained on small datasets.

3.1 The Shape Boltzmann Machine

The SBM used below has two layers of latent variables: $\mathbf{h}^1$ and $\mathbf{h}^2$. The visible units $\mathbf{v}$ are the pixels of a binary image of size $N \times M$. In the first layer we enforce local receptive fields by connecting each hidden unit in $\mathbf{h}^1$ only to a subset of the visible units, corresponding to one of four rectangular patches, as shown in Fig. 3. In order to encourage boundary consistency each patch overlaps its neighbor by r pixels and so has side lengths of $N/2 + r/2$ and $M/2 + r/2$. We furthermore share weights between the four sets of hidden units and patches, however the visible biases b_i are not shared.

Similar constraints have previously been used in the literature (e.g. Desjardins and Bengio 2008; Raina et al. 2009; Lee te al. 2009; Norouzi et al. 2009; Ranzato et al. 2010, 2011), especially in convolutional and tiled-convolutional formulations of RBMs and DBNs. In comparison, in the SBM the receptive field overlap of adjacent groups of hidden units is particularly small compared to their sizes.

Overall, these modifications reduce the number of first layer parameters by a factor of about 16 which reduces the amount of data needed for training by a similar factor. At the same time these modifications take into account two important properties of shapes: first, the restricted receptive field size reflects the fact that the strongest dependencies between pixels are typically local, while distant parts of an object often vary more independently (the small overlap allows boundary continuity to be learned primarily at the lowest layer); second, weight sharing takes account of the fact that many generic properties of shapes (e.g. smoothness) are independent of the image position.

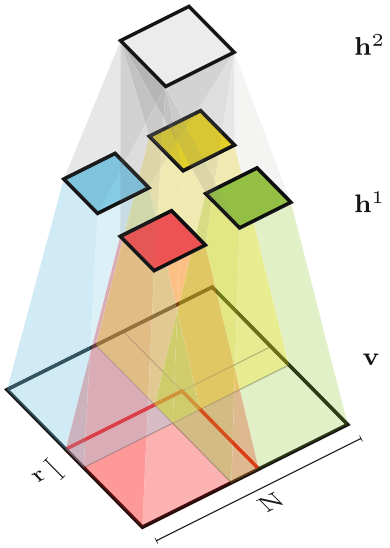

Fig. 3 The Shape Boltzmann Machine in 2D. We enforce local receptive fields by connecting each hidden unit in $\mathbf{h}^1$ only to one of four rectangular patches

For the second layer we choose full connectivity between $\mathbf{h}^1$ and $\mathbf{h}^2$, but restrict the relative capacity of $\mathbf{h}^2$: we use around 4×500 hidden units for $\mathbf{h}^1$ versus around 50 for $\mathbf{h}^2$ in our single class experiments. While the first layer is primarily concerned with generic, local properties, the role of the second layer is to impose global constraints, e.g. with respect to the class of an object shape or its overall pose. The second layer mediates dependencies between pixels that are far apart (not in the same local receptive field), but these dependencies will be weaker than between nearby pixels that share first-level hidden units. Limiting the capacity of the second-layer encourages this separation of concerns and helps to prevent the model from overfitting to small training sets. Note that this is in contrast to Salakhutdinov and Hinton (2009) who use a top-most layer that is at least as large as all of the preceding layers.

3.2 A Multi-region SBM

The SBM model described in the previous section represents shapes as binary images and can be used, for example, as a prior when segmenting a foreground object from its

background. While it is often sufficient to consider the foreground object as a single region without internal structure, there are situations where it is desirable to explicitly model multiple, dependent regions, e.g. in order to decompose the foreground object into parts (Winn and Jojic. 2005; Kapoor 2006; Thomas et al. 2009; Bo and Fowlkes 2011; Eslami and Williams 2011).

In the SBM this can be achieved by using categorical visible units instead of binary ones: visible units with $L+1$ different states (i.e. $v_i \in \{0, \dots L\}$) allow the modeling of shapes with L parts. The visible unit representing the i th pixel then indicates which of the L parts or the background the pixel belongs to (here we treat the background as part 0).

We use a 'one-of-$L+1$' encoding for v_i, i.e. we choose v_i to be $L+1$ dimensional binary vectors and for $v_i = l$ we set $v_{il} = 1$ and $v_{il'} = 0, \quad \forall l' \neq l$. The energy function of this extended model is given by:

$$E(\mathbf{V}, \mathbf{h}^1, \mathbf{h}^2 | \boldsymbol{\theta}^s) = \sum_{i,l} b_{li} v_{li} + \sum_{i,j,l} w^1_{lij} v_{li} h^1_j + \sum_j c^1_j h^1_j + \sum_{j,k} w^2_{jk} h^1_j h^2_k + \sum_k c^2_k h^2_k, \tag{14}$$

where we use $\mathbf{V}$ to denote the the matrix with the $L+1$ dimensional vectors $\mathbf{v}_i$ in its rows.

This change in the nature of the visible units preserves all of the appealing properties of the SBM. In particular the conditional distributions over the three sets of variables $\mathbf{V}$, $\mathbf{h}^1$, and $\mathbf{h}^2$ remain factorial. The only change is in the specific forms of the two conditional distributions $p(\mathbf{v}|\mathbf{h}^1)$ and $p(\mathbf{h}^1|\mathbf{v}, \mathbf{h}^2)$:

$$p(v_i = l | \mathbf{h}^1) = \frac{\exp\left(\sum_j w^1_{lij} h^1_j + b_{li}\right)}{\sum_{l'=0}^{L} \exp\left(\sum_j w^1_{l'ij} h^1_j + b_{l'i}\right)}, \tag{15}$$

$$p(h^1_j = 1 | \mathbf{V}, \mathbf{h}^2) = \sigma\left(\sum_{i,l} w^1_{lij} v_{li} + \sum_k w^2_{jk} h^2_k + c^1_j\right) \tag{16}$$

where in the left-hand-side of Eq. 15 we use $v_i = l$ to denote the fact that $v_{il} = 1$ and $v_{il'} = 0, \quad \forall l' \neq l$ as explained above.

Note that Eq. 16 is effectively the same as Eq. 13 except that there are now $L+1$ binary visible units per pixel. The conditional distribution given in Eq. 15 implements the constraint that for each pixel only one of these $L+1$ binary units can be active, i.e. only one of the parts can be present. Due to the particular form of the conditional distribution (Eq. 15) categorical visible units are often referred to as 'softmax' units (e.g. Bridle 1990). In our experiments below we explore SBMs with six or seven parts.

It should be noted that the above formulation of the multi-part SBM is especially suited to model the shapes of several *dependent* regions such as non-occluding (or lightly occluding) object parts. For modeling the shapes of multiple *independent* regions, as arise in the case of multiple occluding objects, it might be more suitable to model occlusion explicitly, as in Le Roux et al. (2011).

4 Learning

Learning of the model involves maximizing $\log p(\mathbf{v}; \Theta)$ of the observed data $\mathbf{v}$ with respect to its parameters $\Theta = \{\mathbf{b}, W^1, W^2, \mathbf{c}^1, \mathbf{c}^2\}$ (see Eqs. 5, 10). The gradient of the log-likelihood of a single training image with respect to the parameters is given by:

$$\nabla_\Theta \log p(\mathbf{v}; \Theta) = \langle \nabla_\Theta E(\mathbf{v}', \mathbf{h}^1, \mathbf{h}^2; \Theta) \rangle_{p_\Theta(\mathbf{v}', \mathbf{h}^1, \mathbf{h}^2)} - \langle \nabla_\Theta E(\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2; \Theta) \rangle_{p_\Theta(\mathbf{h}^1, \mathbf{h}^2 | \mathbf{v})}, \tag{17}$$

and the total gradient is obtained by summing the gradients of the individual training images (e.g. Ackley et al. 1985; Freund and Haussler 1994; Salakhutdinov and Hinton 2009). The first term on the right hand side is the expectation of the gradient of the energy (see Eqs. 9, 14) where the expectation is taken with respect to the joint distribution over $\mathbf{v}, \mathbf{h}^1, \mathbf{h}^2$ defined by the model. The second term is also an expectation of the gradient of the energy, but this time taken with respect to the posterior distribution over $\mathbf{h}^1, \mathbf{h}^2$ given the observed image $\mathbf{v}$. Although the gradient is readily written out, maximization of the log-likelihood is difficult in practice. Firstly, except for very simple cases it is intractable to compute as both expectations involve a sum over a number of terms that is exponential in the number of variables (visible and hidden units). Secondly, gradient ascent in the likelihood is prone to getting stuck in local optima.

In this work we closely follow the procedure proposed in Salakhutdinov and Hinton (2009) which minimizes these difficulties in three ways: (a) it approximates the first expectation in Eq. 17 with samples drawn from the model distribution via MCMC; (b) it approximates the second expectation using a mean-field approximation to the posterior; and (c) it employs a pre-training strategy that provides a good initialization to the weights W^1, W^2 before attempting learning in the full model.

Learning proceeds in two phases. In the *pre-training* phase we greedily train the model bottom up, one layer at a time. The purpose of this phase is to find good initial values for all parameters of the model. We begin by training an RBM on the observed data. The likelihood gradient of an RBM takes a form very similar to Eq. 17. Unlike for the DBM, for an RBM the second expectation over the conditional distribution of the hidden units $\mathbf{h}$ given the data is tractable and can be computed exactly (see Eq. 8). The first expectation, taken with respect to the full model distribution, however, remains intractable. We therefore perform stochastic maximum likelihood learning

(SML, also referred to as 'persistent contrastive divergence'; Neal 1992; Tieleman 2008; Salakhutdinov and Hinton 2009) where this expectation is approximated using samples from the model distribution obtained via MCMC. While a naïve MCMC approximation of the expectation would be computationally very expensive, considerable computational savings can be obtained through a set of Markov chains that are initialized at the beginning of learning and then maintained over the course of learning (hence the adjunct 'persistent'), alternating updates of the model parameters Θ with Gibbs sampling steps to update the sample approximation to the model distribution. This algorithm is an instance of a stochastic approximation scheme of the Robbins–Monro type (Robbins and Monro 1951; Younes and Sud 1989; Younes 1999).

The number of hidden units of this RBM is the same as the size of $\mathbf{h}^1$ in the full SBM model and it obeys the same connectivity constraints as the SBM's first layer. Once this RBM is trained, we infer the conditional mean of the hidden units using Eq. 8 for each training image. The resulting vectors then serve as the training data for a second RBM with the same number of hidden units as $\mathbf{h}^2$, which is trained using SML.

We use the parameters of these two RBMs to initialize the parameters of the full SBM model as described in Salakhutdinov and Hinton (2009). Simply speaking, we use the weights of the first RBM to initialize the parameters of the lower layer of the SBM ($\mathbf{b}$ and W^1), and the parameters of the second RBM to initialize the upper layer (W^2 and $\mathbf{c}^2$). As discussed in detail in Salakhutdinov and Hinton (2009) special care must be taken to account for the fact that in the full model $\mathbf{h}^1$ now receives input from both $\mathbf{v}$ and $\mathbf{h}^2$.

In the second phase we then perform approximate stochastic gradient ascent in the likelihood of the full model to fine-tune the parameters in an expectation-maximization-like scheme. This involves the same sample-based approximation to the gradient of the normalization constant used for learning the RBMs (Tieleman 2008; Salakhutdinov and Hinton 2009), as well as a fully factorized mean-field approximation to the posterior $p(\mathbf{h}^1, \mathbf{h}^2|\mathbf{v})$. This joint training is essential to separate out learning of local and global shape properties into the two hidden layers.

5 Experiments

We performed an extensive experimental evaluation of the SBM model on five datasets in total. The presentation of the results is divided into four parts:

In Sect. 5.1 we focus on demonstrating that the SBM can indeed act as a strong model of object shape. For this purpose we perform qualitative and quantitative evaluations on two challenging datasets: the Weizmann horse datasets and motorbikes from Caltech-101. Despite both datasets being relatively small we find that the learned models capture essential high- and low-level properties of the shapes in the training data, producing realistic samples and generalizing to novel shapes not present in the training data. Quantitatively we find that the SBM outperforms several baseline models in a difficult shape completion task.

The goal of Sect. 5.2 is to examine the contribution of the various architectural choices detailed in Sect. 3 to the success of the SBM. We address the impact of localized receptive fields, weight-sharing, and of the hierarchical structure of the model.

In many situations it is desirable or even necessary to model not just a single but multiple object classes with the same model. In Sect. 5.3 we therefore introduce an additional dataset comprised of multiple object categories (Weizmann horses and several animals from Caltech-101) and demonstrate that the SBM, with a single set of parameters, can learn a joint model of several categories from unlabeled data, generalizing reliably within each category.

Finally, in Sect. 5.4 we analyze the behavior of the multi-part extension of the SBM introduced in Sect. 3.2 on two multi-part datasets, the ETHZ cars dataset and the HumanEva pedestrians dataset.

5.1 Generalization and Realism

In this section we demonstrate that the SBM can be trained to be a strong model of object shape. For this purpose we consider two challenging datasets: Weizmann horses and Caltech-101 motorbikes.

Weizmann horse dataset The Weizmann horse dataset (Borenstein et al. 2004) contains 327 images, all of horses facing to the left, but in a variety of poses.[1] The dataset is challenging because in addition to their overall pose variation, the positions of the horses' heads, tails and legs change considerably from image to image.

The binary images are cropped and normalized to 32×32 pixels (see Fig. 5a). We trained an SBM with overlap $r = 4$, and 2,000 and 100 units for $\mathbf{h}^1$ and $\mathbf{h}^2$ respectively. The first layer was pre-trained for 3,000 epochs (iterations) and the second layer for 1,000 epochs. After pre-training, joint training was performed for 1,000 epochs. Our MATLAB implementation completed training in around 4 h, running on a dual-core, 3 GHz PC with 4GB of memory.

Caltech motorbikes dataset Our second dataset is based on Caltech-101 (Fei-Fei et al. 2004), and consists of 798 motorbike silhouettes.[2] These binary images are of higher reso-

[1] http://msri.org/people/members/eranb.

[2] http://vision.caltech.edu/Image_Datasets/Caltech101.

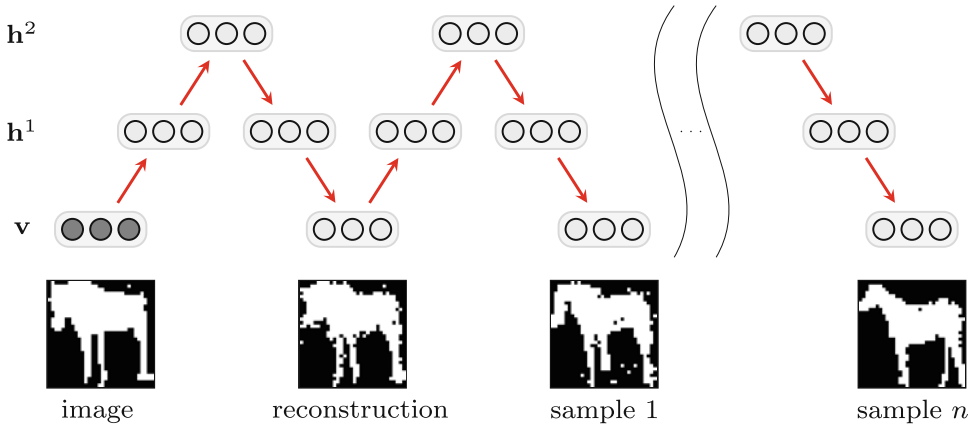

Fig. 4 DBM MCMC. Block-Gibbs MCMC sampling scheme, in which $\mathbf{v}$, $\mathbf{h}^1$ and $\mathbf{h}^2$ variables are sampled in turn. Note that each sample of $\mathbf{h}^1$ is obtained conditioned on the current state of $\mathbf{v}$ and $\mathbf{h}^2$. For sufficiently large values of n, sample n will be uncorrelated with the original image

lution than the horses and are cropped and normalized to 64×64 pixels (see Fig. 7a). We trained an SBM with overlap $r = 4$, and 1,200 and 50 units for $\mathbf{h}^1$ and $\mathbf{h}^2$ respectively, using the same schedule as before.

It is noteworthy that for both datasets the number of training images is relatively small compared to the variability present in the data and, in particular, compared to the size of datasets that deep learning models are typically trained on. Both datasets consist of significantly less than 1,000 training images which is in stark contrast to the several thousand or, more often, tens of thousands of training images for most applications of deep models in the literature. Salakhutdinov and Hinton (2009), for instance, use the 60,000 training images from the MNIST dataset for their experiments.

Baseline models For comparison we considered two baseline models: First, we trained a factor analysis (FA) model with 10 latent dimensions. The FA model was modified to work on discrete binary images. Similar to the clipped factor analysis model described in Cemgil et al. (2005) the independent Gaussian latent variables are mixed linearly and then passed through a sigmoid to obtain binary observed variables:

$$p(\mathbf{h}) = \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{18}$$

$$p(v_i = 1|\mathbf{h}) = \sigma\left(\sum_j w_{ij} h_j + b_j\right), \tag{19}$$

where $\mathbf{0}$ is a vector of zeros and $\mathbf{I}$ denotes the identity matrix. The model was trained using gradient ascent, and inference was performed using elliptical slice sampling as described in Eslami and Williams (2011).

Our second baseline model was the RBM as defined in Eq. 4. We used 500 hidden units and trained the model using SML as described in Sect. 4. For both baseline models the hyperparameters and number of hidden units were manually optimized for each dataset.

5.1.1 Realism

To assess the Realism requirement, we sampled a set of shapes from each model, as shown in Figs. 5 and 7 for the horse and motorbike datasets respectively.

The FA shape models can be sampled from directly. For the RBM and SBM models samples are generated by extended block Gibbs sampling. In particular, for the SBM models samples were generated using the scheme outlined in Fig. 4. As is common in the literature, we visualize the samples by showing for each pixel i the (grayscale) *conditional probability* of that pixel $p(v_i = 1|\mathbf{h})$ given the particular hidden configuration that constitutes the current state of the Markov chain. Binary samples can be generated per-pixel from a Bernoulli distribution where the gray level specifies the distribution mean.

FA effectively defines a transformed Gaussian distribution over the image pixels and is thus inherently unimodal. In order to account for the diversity of shapes in the training data it is therefore forced to allocate probability mass to images that do not correspond to realistic horse or motorbike shapes, as shown in Figs. 5b and 7b.

By contrast, the RBM can, in principle, account for multimodal data and could thus assign probability mass more selectively. However, as the samples of horses (Fig. 5c) indicate, the model also fails to learn a good model of the variability of horse shapes—the samples are mostly of the same pose, and details of the shape are lost when the pose changes. We found this effect to be even more dramatic for RBM samples of motorbikes, due to the larger image size (see Fig. 7c).

These problems are symptomatic of training RBMs with insufficient data. The SBM aims to overcome these problems through a combination of connectivity constraints, weight sharing, and model hierarchy. As we will discuss in more detail in Sect. 5.2 below, the combination of these ingredients is necessary to obtain a strong model of shape.

Samples from the SBM for horses and motorbikes are shown in Figs. 5d and 7d respectively. First, we note that

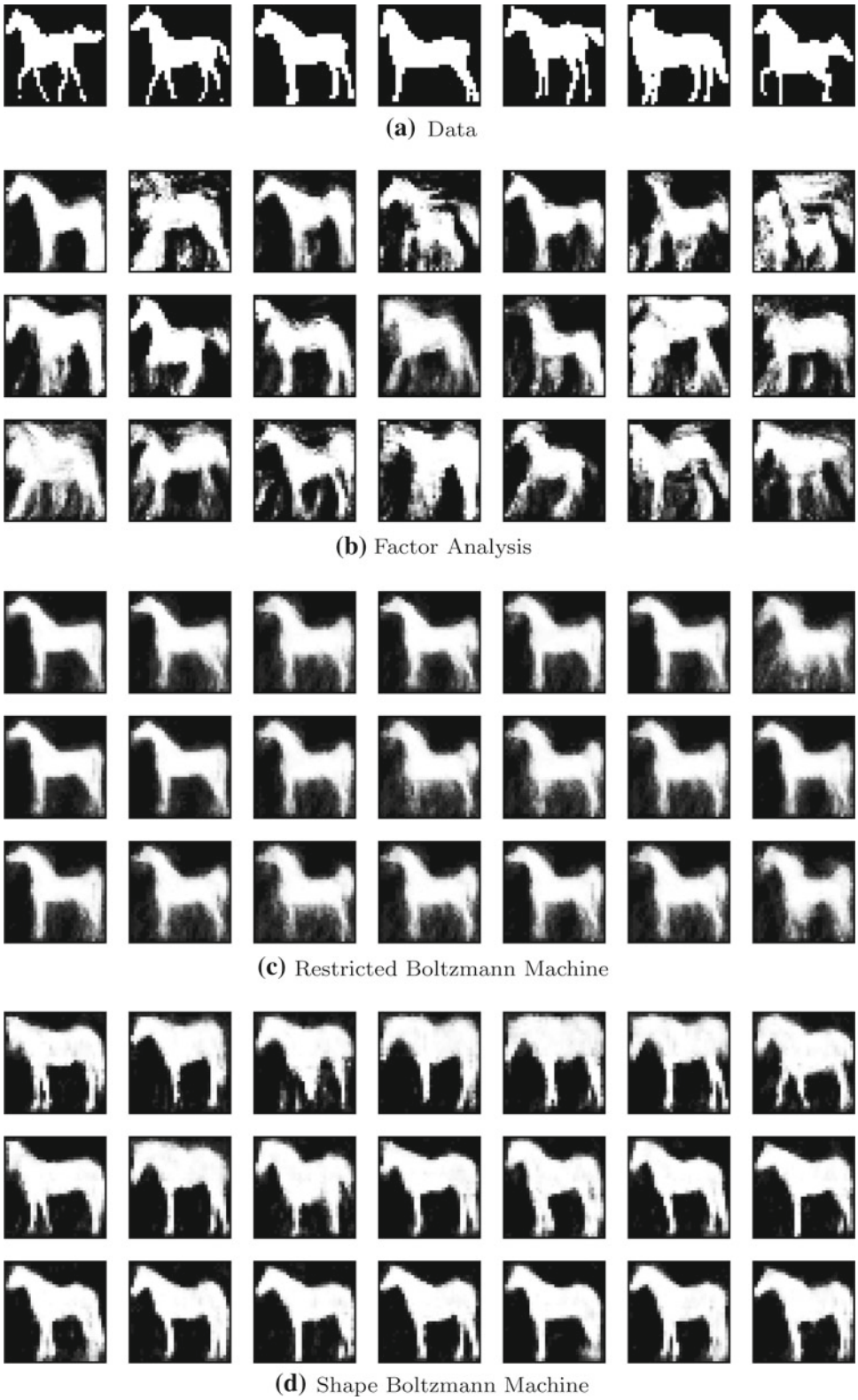

Fig. 5 Sampled horses. (**a**) A selection of images from the Weizmann horse dataset, (**b**) A collection of samples from a discrete factor analysis model. The Gaussianity assumption forces the model to allocate probability mass to unlikely horse shapes, (**c**) Samples from an RBM, (**d**) Samples from an SBM. The model generates samples of varying pose, with the correct numbers of legs and details are preserved (samples are arranged *left-right*, *up-down* in decreasing order of generalization)

the model generates natural shapes from a variety of poses. Second, we observe that details such as legs (in the case of horses) or handle bars, side mirrors, and forks (in the case of motorbikes) are preserved and remain sharply defined in the samples. Third, we note that the horses have the correct number of legs while motorbikes have, for instance, the

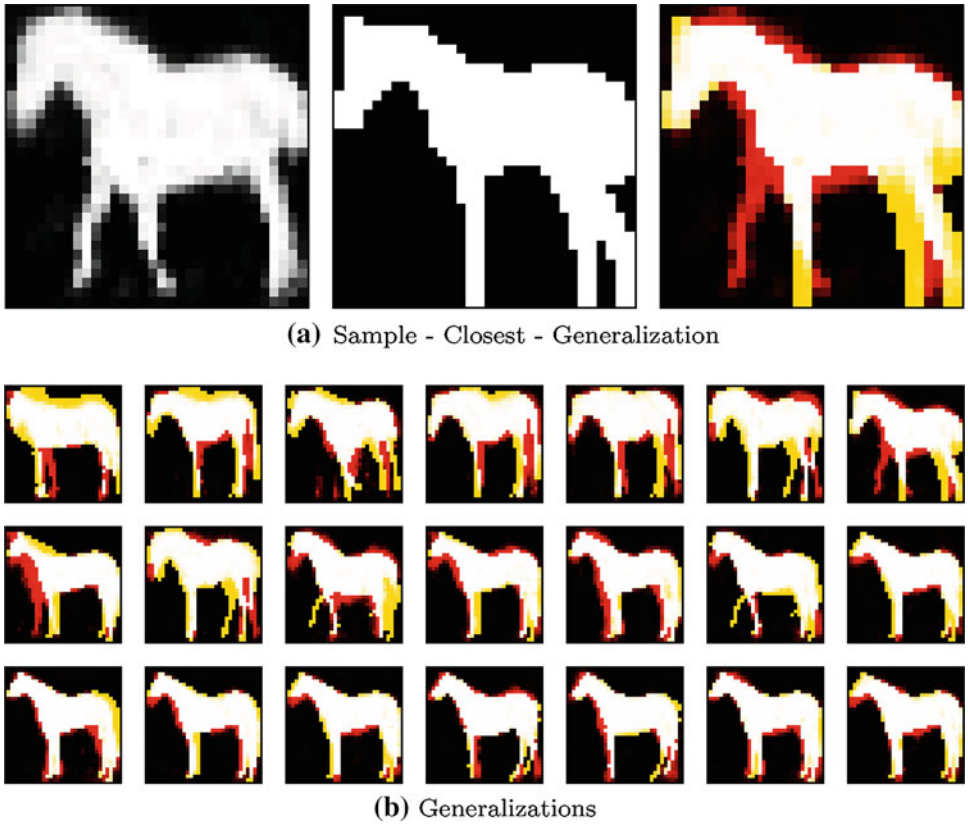

Fig. 6 Generalization. (**a**) A sample from the SBM, the closest image in the training dataset to the generated sample, and the difference between the two images. *Red* pixels have been generated by the sample but are absent in the training image; *yellow* pixels are present in the training image but absent in the sample. The model has generalized to an unseen, but realistic horse shape, (**b**) Generalizations made in each of the samples in Fig. 5d

correct number of handle bars and wheels. Finally, we note that the patch overlap ensures seamless connections between the four quadrants of the image. Indeed, horse and motorbike samples generated by the model look sufficiently realistic that we consider the model to have fulfilled the Realism requirement.

5.1.2 Generalization

We next investigated to what extent the SBM meets the generalization requirement, to ensure that the model has not simply memorized the training data. In Fig. 6 we show for horses the difference between the sampled shapes from Fig. 5d and their closest images in the training set. We use the Hamming distance between training images and a thresholded version of the conditional probability (>0.3), as the similarity measure. This measure was found to retrieve the visually most similar images. Red indicates pixels that are in the sample but not in the closest training image, and yellow indicates pixels in the training image but not in the sample. Fig. 7e shows a similar analysis for samples from the model learned for motorbikes. Both models generalize from the training datapoints in non-trivial ways whilst maintaining validity of the overall object shape. These results suggest that the SBM generalizes to realistic shapes that it has not encountered in the training set.

5.1.3 Shape completion

We further assessed both the realism and generalization capabilities of the SBM by using it to perform shape completion, where the goal is to generate likely configurations of pixels for a missing region of the shape, given the rest of the shape. To perform completion we obtain samples of the missing—or unobserved—pixels $\mathbf{v}_U$ conditioned on the remaining (observed) pixels $\mathbf{v}_O$ (U and O denote the set indices of unobserved and observed pixels respectively). This is achieved using a Gibbs sampling procedure that samples from the conditional distribution. In this procedure, samples are obtained by running a Markov chain as before, sampling $\mathbf{v}$, $\mathbf{h}^1$, and $\mathbf{h}^2$ from their respective conditional distributions, but every time $\mathbf{v}$ is sampled we 'clamp' the observed pixels $\mathbf{v}_O$ of the image to their given values, updating only the state of the unobserved pixels $\mathbf{v}_U$. Since the model specifies a *distribution* over the missing region $p(\mathbf{v}_U|\mathbf{v}_O)$, multiple such samples capture the variability of possible solutions that exist for any given completion task. In Fig. 8 we show how the samples become more constrained as the missing region shrinks. Figures 9 and 10 show sampled completions of regions of horse and motorbike images that the model had not seen during training. Despite the large sizes of the missing portions, and the varying poses of the horses and motorbikes, completions look realistic.

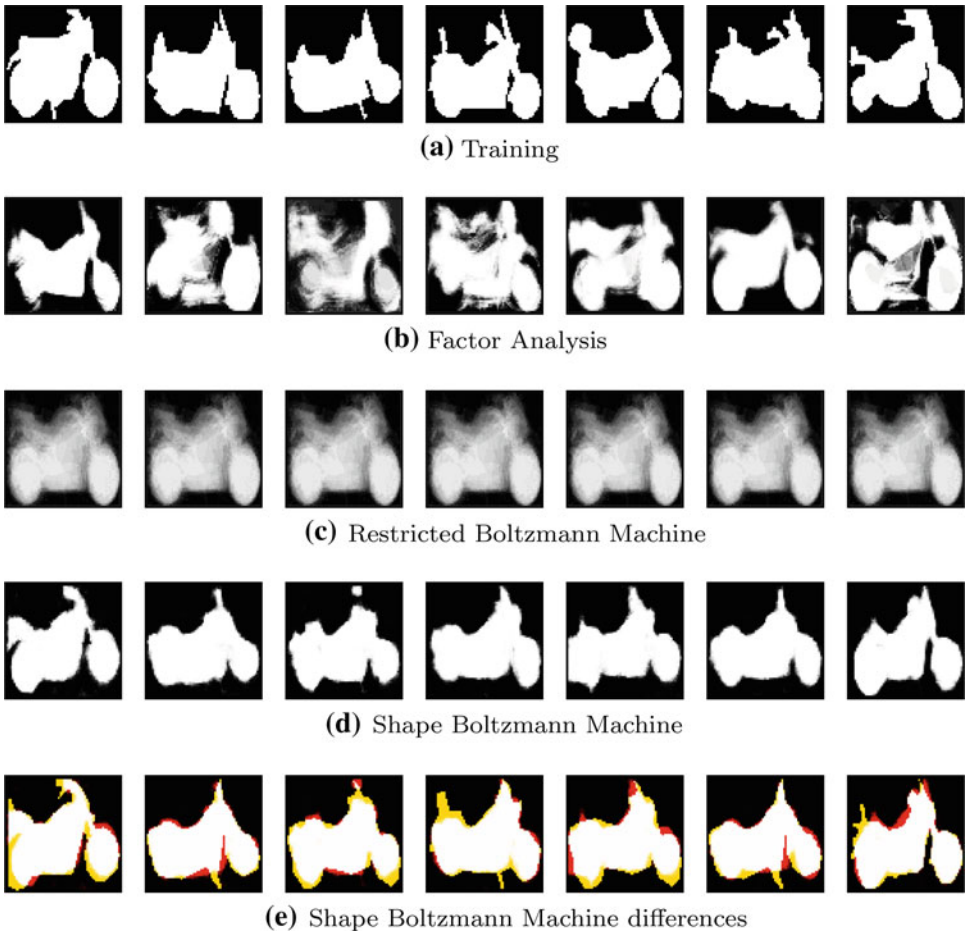

Fig. 7 Results on Caltech-101 motorbikes. (**a**) A selection of images from the training set (at 64×64 pixels), (**b**) A set of samples from the FA baseline model, (**c**) A set of samples from the RBM baseline model, (**d**) A chain of samples generated by the SBM, (**e**) Difference images for each of the samples in (**d**) (same format as in Fig. 6): the model generalizes from training examples in non-trivial ways, whilst maintaining overall motorbike look-and-feel

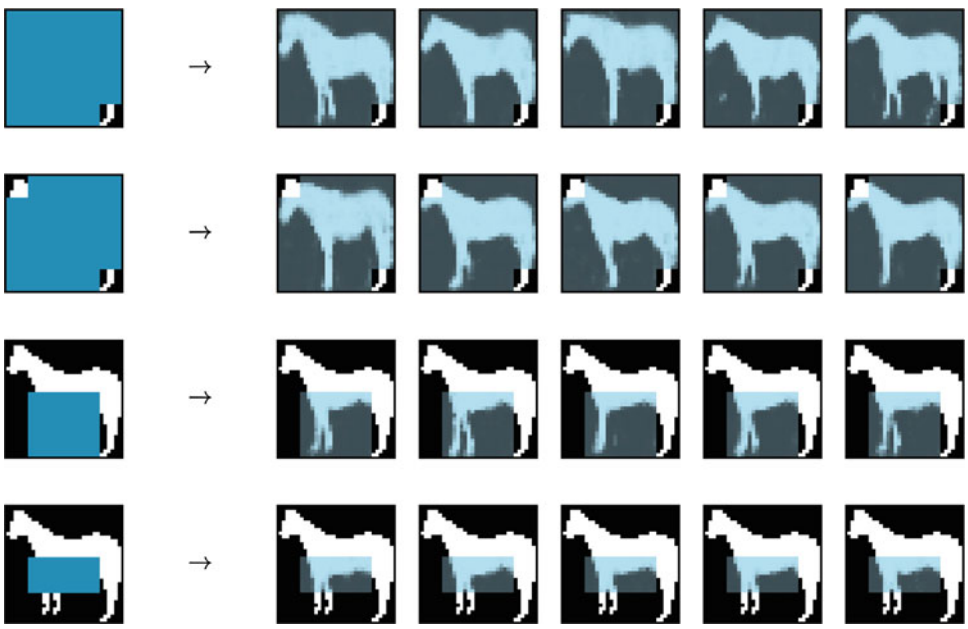

Fig. 8 Shape completion variability. *Blue* in the first *column* indicates the missing regions. The samples highlight the *variability* in possible completions captured by the model. As the missing region shrinks, the samples become more constrained

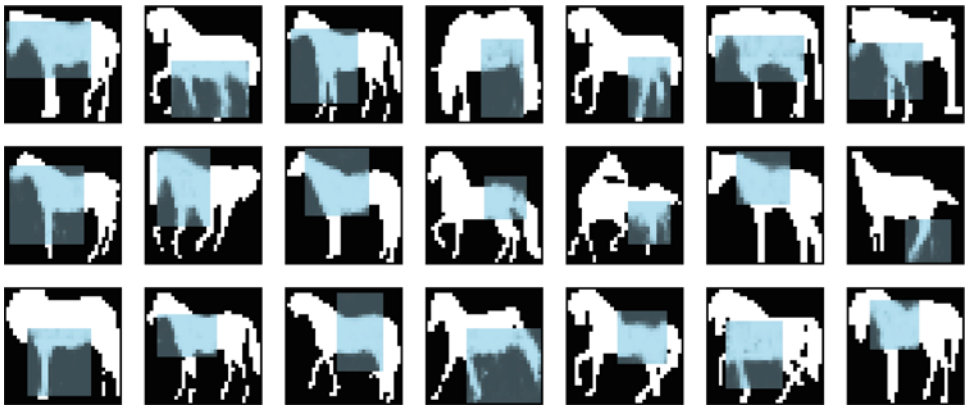

Fig. 9 Sampled image completion for horses. The SBM completes rectangular imputations of random size on images not seen during training

Fig. 10 Sampled image completion for motorbikes

The SBM's ability to do shape completion suggests applications in a computer graphics setting. Sampled completions can be constrained in real-time by simply clamping certain pixels of the image. In Fig. 11a and c we show snapshots of a graphical user interface in which the user modifies a horse or motorbike silhouette with a digital brush. The model's ability to generalize enables it to generate samples that satisfy the user's constraints. The model's accurate knowledge about horse and motorbike shapes ensures that the samples remain realistic.

As a direct comparison we also consider a simple database driven ('non-parametric') approach where we try to find suitable completions via a nearest-neighbor search in our database of training shapes. As shown in Fig. 11 such a database-driven approach can fail to find shapes that match the constraints.

The same approach can also be used to generate complete silhouettes in different poses given simple stick figures provided by the user (see Fig. 11b, d). This GUI and a video showing its use may be downloaded from http://bit.ly/ShapeBM.

5.1.4 Quantitative Comparison

A natural way to directly evaluate a generative model *quantitatively* is by computing the likelihood of some held-out data under the model. Unfortunately, this likelihood computation is intractable for DBMs. Approximations, e.g. based on annealed importance sampling, (Neal 2001; Salakhutdinov and Murray 2008; Salakhutdinov and Hinton 2009; Murray and Salakhutdinov 2009) are computationally very expensive and their accuracy can be difficult to assess.

As an alternative we therefore introduce what we will refer to as an 'imputation score' for the shape completion task as a measure of the strength of a model. We collect additional horse and motorbike silhouettes from the web (25 horses and 25 motorbikes), and divide each into nine segments. We then perform multiple imputation tests for each image. In each test, we remove one of the segments and estimate the conditional probability of that segment under the model, given the remaining eight segments. The log probabilities are then averaged across the different segments and images to give the score.

Except for the mean model (where they are trivial) the conditional distributions over the subsets of unobserved pixels given the rest of the image are infeasible to compute in practice due to the dependencies introduced by the latent variables. We therefore approximate the required conditional log-probabilities via MCMC: for a particular image and segment we draw configurations of the latent variables from the posterior given the observed part of the image and then evaluate the conditional probability of the true configuration of the unobserved segment given the latent variables, i.e. we compute:

$$p(\mathbf{v}_U|\mathbf{v}_O) \approx \frac{1}{S}\sum_s p(\mathbf{v}_U|\hat{\mathbf{h}}^s), \tag{20}$$

where $\mathbf{v}_U$ and $\mathbf{v}_O$ indicate the set of unobserved/observed pixels (corresponding to the one removed and the eight remaining segments), and $\hat{\mathbf{h}}^s \sim \mathbf{h}|\mathbf{v}_O$ are samples from the conditional distribution over the hidden units given the observed part of the image obtained via MCMC.[3] Provided that our MCMC scheme allows us to sample from the true posterior the right hand side of Eq. 20 provides us with an unbiased estimate of $p(\mathbf{v}_U|\mathbf{v}_O)$.

[3] We set $S = 10,000$ in our experiments.

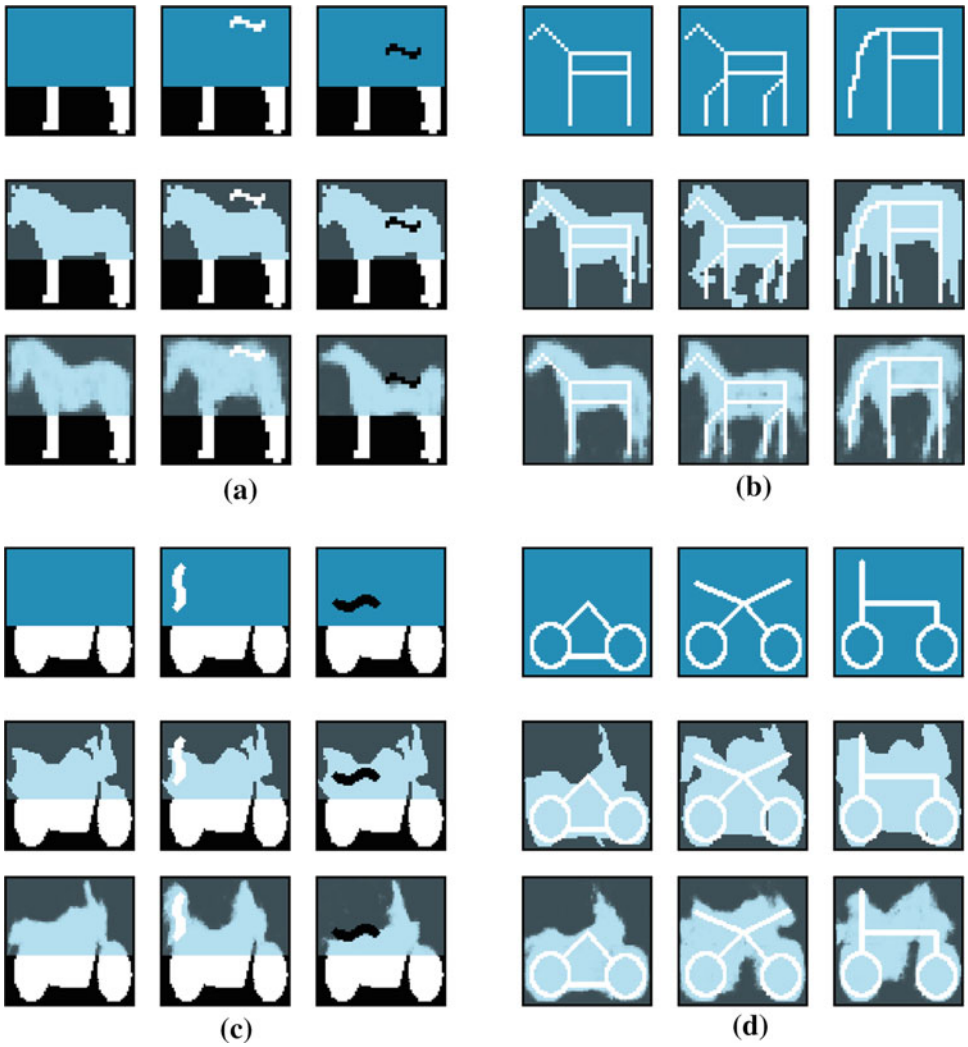

Fig. 11 Constrained shape completion. Missing regions (*blue* pixels, *top row*) are completed using the SBM and by finding the closest match (*middle row*) to the prescribed pixels in the training data. (**a**) The horse's back is pulled up by the SBM (*bottom row*) using an appropriate 'on' brush. Notice how the stomach moves up and the head angle changes to maintain a valid shape. The horse's back is then pushed down with an 'off' brush, (**b**) Given only minimal user input, the model completes the images to generate realistic shapes. (**c**), (**d**) Motorbikes. In many cases, the nearest neighbor method fails to find a suitable training image to satisfy the constraints

A high score in this test indicates both the realism of samples and the generalization capability of a model, since models that do not allocate probability mass on good shapes (from the 'true' generating distribution of horses) and models that waste probability mass on bad shapes are both penalized. In particular for the motorbike dataset we found a small amount of regularization to be beneficial for most models. This prevented overly confident predictions (and hence large penalties in the log-probability), e.g. in the situation where a particular pixel happened to be 0 for all training images, but 1 in one or some of the test images. To this end we replaced the predicted probability p of a pixel being 1 given the observed portion of the image by $d + (1 - 2d) \cdot p$. The results of these experiments can be seen in Table 2. For optimal damping SBM is the top-performing model on both the horses and motorbikes datasets, but the FA model performs well on the motorbikes.

Table 2 Imputation scores

	Horses Score	d	Motorbikes Score	d
Without regularization				
Mean	−50.72	0.000	−248.28	0.000
FA	−41.28	0.000	**−109.17**	0.000
RBM	−48.57	0.000	−142.47	0.000
SBM	**−27.90**	0.000	−132.97	0.000
With regularization				
Mean	−50.65	0.012	−154.14	0.010
FA	−40.33	0.028	−108.41	0.006
RBM	−47.52	0.016	−142.47	0.000
SBM	**−26.90**	0.014	**−104.21**	0.034

In the 'with regularization' scenario, we also report for each model the regularization d which maximizes that model's score
Bold values indicate the highest score achieved by the four models on each dataset in each scenario

5.2 Analysis of the SBM Formulation

So far we have demonstrated that the SBM is able to learn strong models of object shapes, producing realistic samples without overfitting to the training data. In this section we explore in more detail how these capabilities of the SBM depend on the specific properties of the architecture described in Sect. 3: local receptive field and weight sharing; hierarchical formulation; and receptive field overlap.

5.2.1 Generalization Through Local Receptive Fields

In the first layer of the SBM we employ localized receptive fields and parameter sharing. This dramatically reduces the number of parameters that need to be learned and in consequence substantially reduces the propensity of the model to overfit.

One way to diagnose this effect is to inspect the first layer weight matrix of the SBM and compare it to those of the two baseline models (RBM and FA) which were implemented without weight sharing. Each column in the weight matrices W of the models (Eqs. 4, 9, 19 for the RBM, SBM, and FA model respectively) corresponds to a 'filter' that is associated with the activation of one of the hidden units. As shown in Fig. 12a, b, the filters for the FA and RBM have only global structure. This means that these models are unable to combine local filters to generate novel horse shapes. In contrast, because spatial locality and parameter-sharing are built into the SBM, it learns general-purpose filters that allow it to generalize factorially from the training examples as can be seen in Fig. 12c.

Increasing the number of hidden units in the RBM in the hope that additional capacity would allow it to learn more local filters did not solve the problem but rather worsened the overall results, suggesting that it is indeed the lack of data rather than a lack of capacity that is the issue. On the other hand, an RBM with similar connectivity constraints as the first layer of the ShapeBM has fewer parameters than a fully connected RBM and thus suffers less from overfitting (cf. Fig. 13). But as we discuss in more detail in the next section without the second layer it fails to account for global constraints on the shape.

5.2.2 Global Consistency Through Hierarchy

Localized receptive fields and weight sharing are crucial for the ability of the SBM to generalize well. In order to obtain a model that produces realistic samples these need to be embedded in a hierarchical architecture that ensures the global consistency of the shapes.

This is demonstrated by the samples in Fig. 13: They are obtained from an RBM equivalent to only the first layer of the SBM, i.e. this RBM has localized receptive fields with a small overlap between them. It was trained on the Weizmann horse dataset and has the same number of hidden units as the first layer of the horse SBM for which we have shown samples above. Unlike the fully connected RBM whose samples are shown in Fig. 5c this constrained RBM learns to generate a diverse set of shapes. The samples are, however, only locally plausible. In contrast to the samples from the SBM they do not exhibit any of the large-scale structure present in the training data and therefore are not realistic horse shapes in most cases.

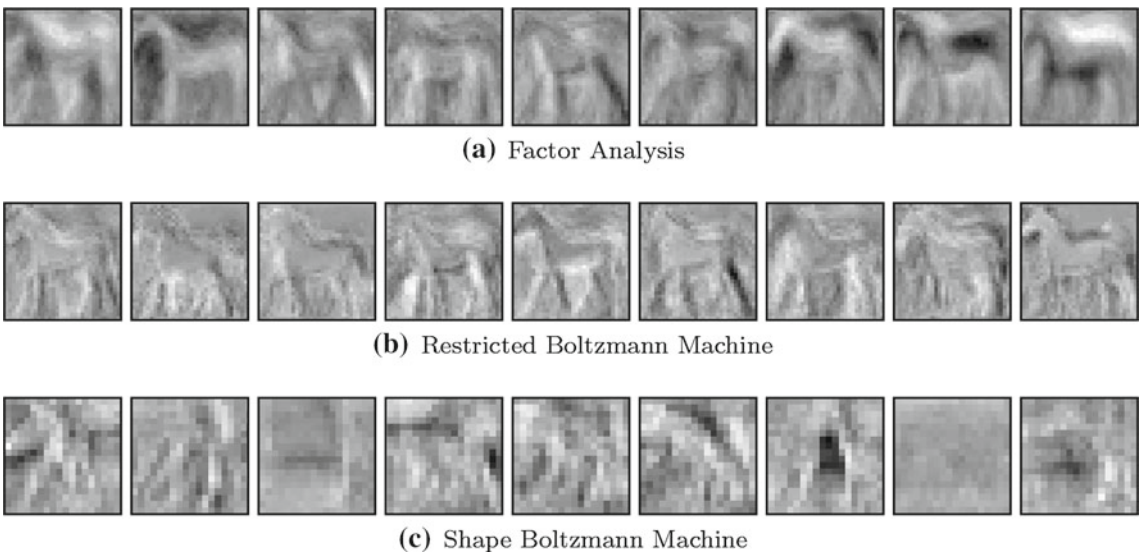

Fig. 12 First layer example weights. (**a**) Weights learned by the FA model capture only global modes of variability (32 × 32), (**b**) Weights learned by the RBM also fail to capture local modes of variation (32 × 32), (**c**) General, more local filters learned by an SBM (18 × 18)

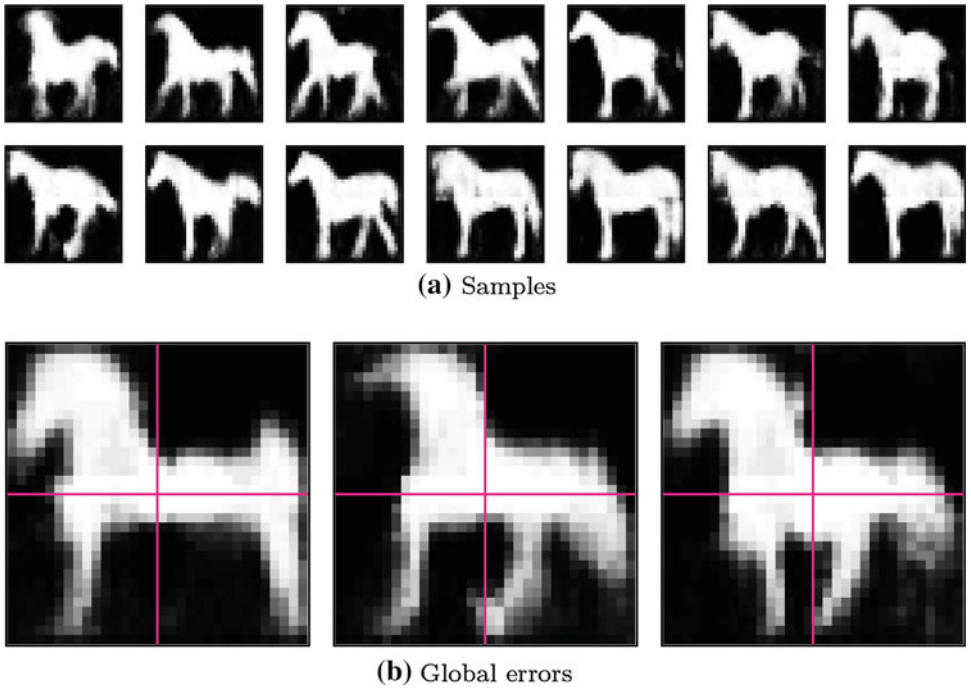

Fig. 13 Samples from an SBM with only a single layer. (**a**) A set of samples drawn from an RBM with the same connectivity constraints (localized receptive fields; small receptive field overlap; weight sharing) as the first layer of the SBM. Although the RBM enforces local smoothness (including at the receptive field boundaries, due to the overlap) it fails to enforce global constraints on the pose of the horses therefore often appears distorted (see, in particular, examples in (**b**); the *pink lines* indicate receptive field boundaries). Note that the visible biases b_i are *not* shared, and this is what allows the model to reproduce very coarsely the main features of horse shapes

Fig. 14 Clamped sampling. Sampling chains are run for two fixed, but different, configurations of $\mathbf{h}^2$. The horse's pose remains fixed, but configurations of legs, and neck and back positions vary

The second layer of the SBM is crucial for enforcing global consistency of the shapes.

In order to further understand the role of the hierarchy and to tease apart the roles of the two layers of the SBM in representing shape information we performed the following experiment: we fixed the configuration of the hidden units in the second layer ($\mathbf{h}^2$) to values inferred from training images and then iterated between sampling $\mathbf{v}$ and $\mathbf{h}^1$ only. In Fig. 14 we plot two sets of samples for two different settings of $\mathbf{h}^2$. We observe that by freezing $\mathbf{h}^2$ we fix the horse's pose, but since $\mathbf{h}^1$ changes from sample to sample the position of its legs and other small details vary. This suggests that the highest layer in the model predominantly captures global information and has learned to be *invariant* to small-scale changes in shape (achieving an effect similar to the pooling layers e.g. in (Lee te al. 2009). This automatic, implicit, separation of large-scale and small-scale statistics is fundamental to the operation of the model.

5.2.3 Local Consistency Through Receptive Field Overlap

The hierarchical formulation encourages *global* consistency of the shapes by coordinating the overall pose across receptive fields. In order to also ensure *local* consistency at the receptive field boundaries we further introduced a small overlap of the receptive fields (denoted by r in Fig. 3).

The effect of this is illustrated in Fig. 15 where we show samples from an SBM (two-layer with local receptive fields and weight sharing) trained in the usual manner, except that there is *no* receptive field overlap (i.e. $r = 0$). This leads to a loss of continuity at the patch boundaries and also (albeit to a lesser extent) to a more global deterioration of sample quality, suggesting that the second layer on its own struggles to enforce local consistency. This global deterioration is due to the fact that some of the modeling capacity of the second layer is now needed to enforce local continuity. Increasing the number of hidden units in the *second* layer would reduce this deterioration at the cost of increasing the number of parameters and so reducing the advantage gained from the hierarchical structure. Experimentally we found that it led to overfitting and did not give satisfactory results.

5.3 Multiple Object Categories

Class-specific shape models are appropriate if the class is known, but for segmentation/detection applications this may

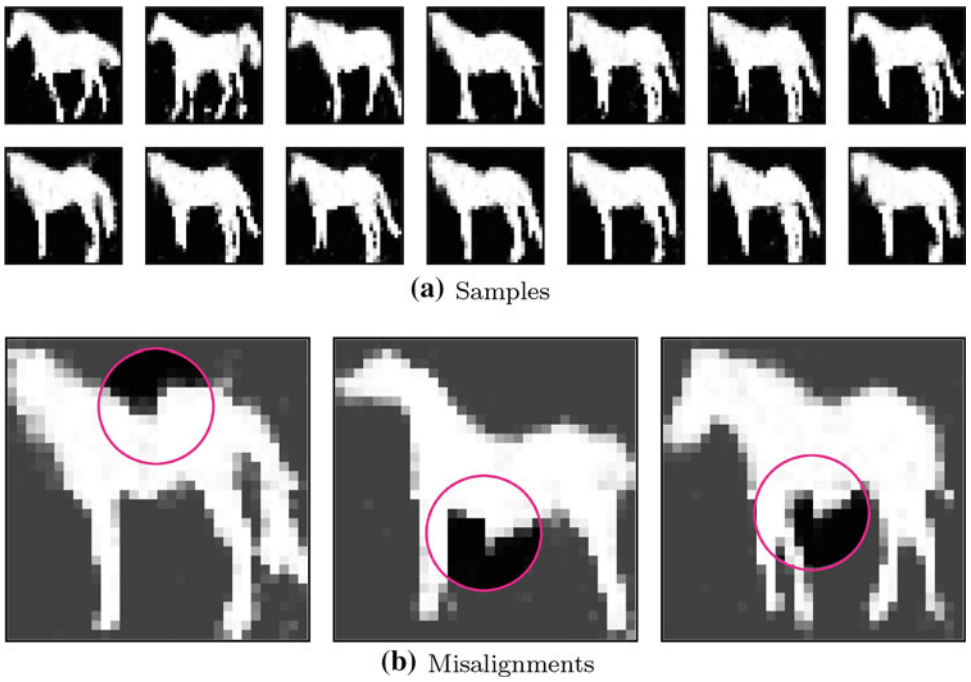

Fig. 15 Samples without overlap. (**a**) Samples from a SBM trained on Weizmann horses in the same way as the SBM described in Sect. 5.1 except that there is no receptive field overlap in the first layer (i.e. $r = 0$). The lack of receptive field overlap leads to discontinuities at the receptive field boundaries not present in the samples from the SBM trained with $r = 4$ (see in particular the examples highlighted in (**b**) and compare to the SBM samples shown in Fig. 5d) and more generally reduces the overall sample quality somewhat

not be the case. A similar situation arises if the view point is not fixed (e.g. objects can appear right or left facing). In both cases there is large overall variability in the data but the data also form relatively distinct clusters of similar shapes (e.g. all objects from a particular category, or all right-facing objects).

To investigate whether the SBM is able to successfully deal with such additional variability and structure in the data we applied it to a dataset consisting of shapes from multiple object classes and tested whether it would be able to learn a strong model of the shapes of all classes simultaneously.

We trained an SBM on a combination of the Weizmann data and three other animal categories from Caltech-101 (Fei-Fei et al. 2004). In addition to 327 horse images, the dataset contains images of 68 dragonflies, 78 llamas and 59 rhinos (for a total of 531 images). The images are cropped and normalized to 32×32 pixels. An SBM with $r = 4$, and 2,000 and 400 units for $\mathbf{h}^1$ and $\mathbf{h}^2$ was jointly trained *without* information about image class.

In our experiments we found that the SBM still learns a strong model, as demonstrated by Fig. 16 which shows samples as well as shape completions obtained from the learned model.

We further wanted to know whether the SBM's unsupervised learning procedure has led it to discover the underlying grouping of the shapes into categories. In order to test this, we compute average inter- and intra-class distances of all training instances, both in data-space ($\mathbf{v}$) and in latent-space ($\mathbf{h}^2$). In Fig. 17a we plot the ratio of these distances for the four classes. These results suggest that the SBM latent representation groups the shapes from each category much more closely than they are in pixel-space.

We also tested how well the model discovered object categories by using it to classify in a setting with very few labeled examples. We trained a generalized linear model (GLM) using the `glmnet` algorithm (Friedman et al. 2010) on between $T = 1 \ldots 20$ randomly selected images of each category and tested on $59 - T$ images per category, averaging over 100 runs. We find that despite its smaller size, given only a few training examples, the latent $\mathbf{h}^2$ is most discriminative (see Fig. 17b). After just one labeled example per category, classification accuracy using the trained GLMs is 56.0% using $\mathbf{h}^2$ versus just 36.8% using $\mathbf{v}$.

Overall these results suggest that the SBM is not only able to deal with the additional variability arising from multiple object classes, but also reliably generalizes within each class. It further appears to naturally separate clusters of related shapes in its latent representation, which can be exploited, for instance, for classification purposes.

5.4 Multiple Object Parts

For the evaluation of the multi-part formulation of the SBM presented in Sect. 3.2 we considered the ground truth label images from two segmentation datasets:

ETHZ cars dataset The first dataset that we considered was the ETHZ labeled cars dataset (Thomas et al. 2009), which itself is a subset of the LabelMe dataset (Russell et al. 2008). It consists of 139 images of cars, all in the same semi-profile

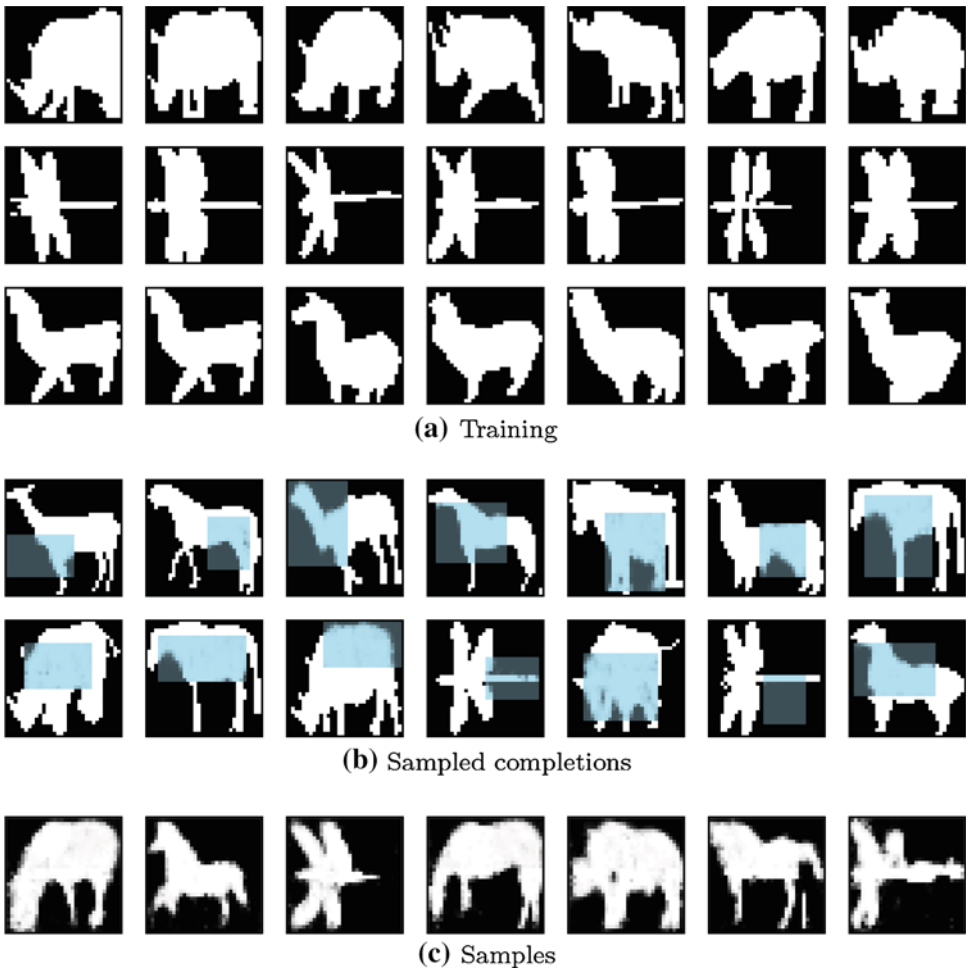

Fig. 16 Multiple object categories. (**a**) A selection of images from the augmented dataset, (**b**) The model simultaneously identifies the object class and fills in the missing image region, (**c**) Samples from a single tempered chain

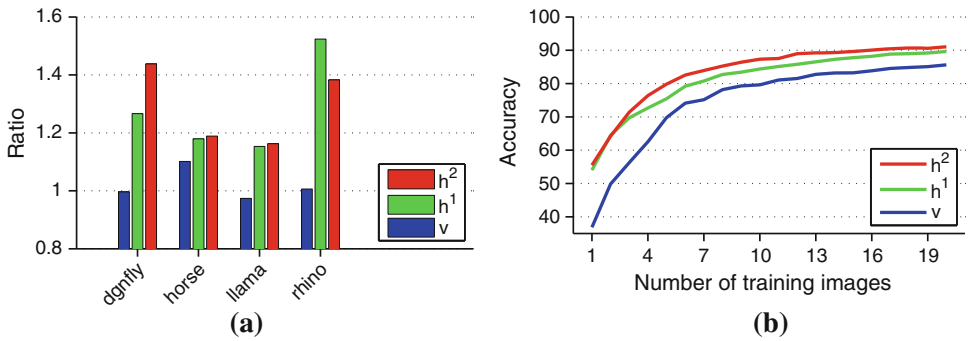

Fig. 17 (**a**) The ratio of inter- and intra-class distances (values >1 indicate that inter-class distances are larger), (**b**) GLM classification accuracy as a function of the number of training images, averaged over 100 runs

view. We used the associated ground-truth segmentations for $L = 6$ parts (body, wheel, window, bumper, license plate, headlight; see Fig. 18a for examples). We trained an SBM at 50×50 pixels with overlap $r = 4$, and 2,000 and 100 hidden units in the first and second layers respectively. Each layer was pre-trained for 3,000 epochs and joint training was performed for 1,000 epochs.

HumanEva pedestrians dataset The second dataset we considered was a labeled version of HumanEva (Sigal et al. 2010; annotations by Bo and Fowlkes 2011) showing humans in different poses and facing in different directions. The images are annotated with ground-truth segmentations for $L = 7$ different parts (hair, face, upper and lower clothes, shoes, legs, arms; see Fig. 19a). We trained an SBM on 684 images together with their flipped counterparts (for a total of 1,368 images) at 48×24 pixels with overlap $r = 4$ (this corresponds to a receptive field size in the first layer of 26×14), and 400 and 50 hidden units in the first and second layers respectively. Each layer was pre-trained for 3,000 epochs. After pre-training, joint training was performed for 1,000 epochs.

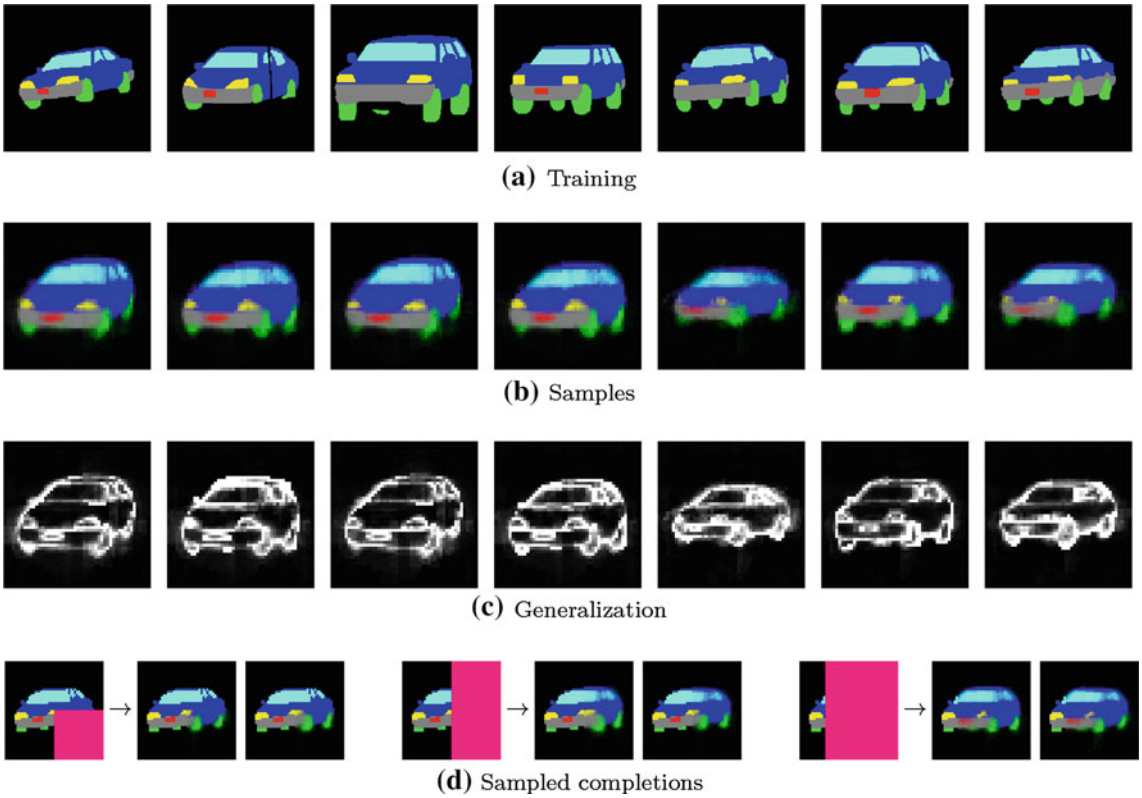

Fig. 18 ETHZ cars. (**a**) Examples from the training data. Different *colors* represent different object parts, (**b**) A chain of samples (1,000 samples between frames). The apparent 'blurriness' of samples is *not* due to averaging or resizing. We display the *probability* of each pixel belonging to different parts. If, for example, there is a 50–50 chance that a pixel belongs to the *red* or *blue* parts, we display that pixel in purple, (**c**) Differences between the samples and their most similar counterparts in the training dataset, (**d**) Sampled completions of occlusions (*pink*). For each occlusion we show two different completions produced by the model (i.e. we show two different samples from the conditional distribution over the unobserved pixels)

To assess the realism and generalization characteristics of the learned SBM models we then performed experiments analogous to the ones in Sect. 5.1: Figures 18b and 19b show a chain of unconstrained samples from the SBM models learned for cars and pedestrians respectively. The models capture highly non-linear dependencies in the data whilst preserving the objects' details (such as face and arms for the pedestrians; or headlights, license plates, and the window frames for cars). We also show for each sample the difference to the closest image in the training set (based on per-pixel label agreement). We see that the model generalizes in non-trivial ways to generate realistic shapes that it had not encountered during training.

We also evaluated the models on constrained shape completion tasks: In Figs. 18d and 19d we show how the SBM completes rectangular occlusions. The left-most example of Fig. 19d highlights the variability in possible completions captured by the model. In the middle example the length of the person's trousers on one leg affects the predictions for the other, demonstrating the model's knowledge about long-range dependencies.

Overall these results demonstrate that the multi-part formulation of the SBM significantly extends the binary SBM in that it allows the modeling of shapes with internal structure while preserving its ability to produce realistic samples and to generalize in a meaningful manner from the training data.

6 Discussion

Thanks to its formulation as a generative model the SBM is very versatile. In our experiments we investigated it as a 'stand-alone' shape model and focused on its ability to generate and complete shapes. But it can also directly be used as a component of a more comprehensive probabilistic architecture: As demonstrated in Le Roux et al. (2011), Heess et al. (2011), Eslami and Williams (2012) and Chen et al. (2013), for instance, it is possible to combine undirected models of shapes formulated as RBMs or DBMs with models of appearance to obtain complete probabilistic generative models of RGB images with well-defined and efficient inference schemes. Such models allow reasoning about var-

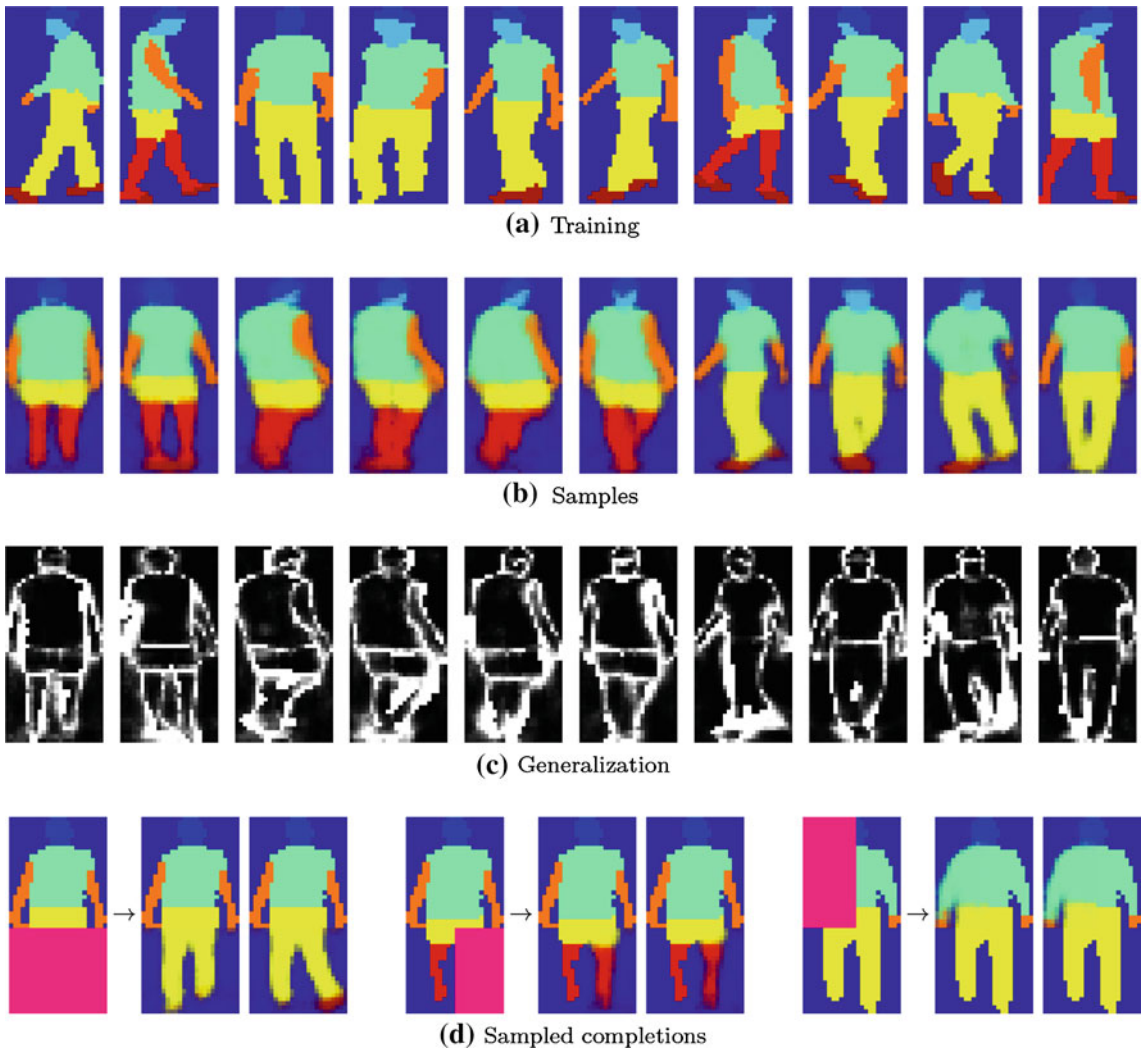

Fig. 19 HumanEva results. (**a**) A selection of images from the dataset, (**b**) A chain of samples (1,000 samples between frames); same format as in Fig. 18, (**c**) Differences between the samples and their most similar counterparts in the training dataset. As observed for the horses and motorbikes the model generalizes in interesting and non-trivial ways to pedestrian shapes not present in the training data, (**d**) Sampled completions of occlusions (*pink*). For each occlusion we show two example completions. Note the variability in the conditional distribution for the large scale occlusion on the *left*

ious image properties and can be applied, for instance, to segmentation tasks. Indeed, Eslami and Williams (2012) use the multi-region SBM presented in Sect. 5.4 to obtain competitive results on two challenging parts-based segmentation benchmarks.

There are three main open questions associated with such applications of the SBM:

Firstly, our shape models are currently of fairly low resolution compared to many real-world images. Naïvely scaling up the SBM by increasing the receptive field size is unlikely to work as this would greatly increase the number of parameters (and hence the potential to overfit) and also lead to practical problems such as slow mixing when sampling from the model. Eslami and Williams (2012) have demonstrated how to side-step these problems by upsampling the predictions of the low-resolution shape prior at test-time. This appears to work well in practice but it still limits the level of detail at which shapes can be modeled.

A second open question is that of translation and scale invariance. These invariances are challenges for many dense, pixel-level models, not just the SBM. Convolutional architectures (e.g. Desjardins and Bengio 2008; Roth and Black 2005; see also e.g. Ranzato et al. 2010) are inherently translation invariant but can be expensive as they require enough capacity to learn the structure of interest at all possible positions. An alternative way to achieve large-scale translation invariance is through a model that is defined only for a tight bounding box enclosing the shape and which is then explicitly translated to all possible image positions (e.g. Frey et al. 2003; Williams and Titsias 2004; similar to the sliding

window approach for object detection e.g. Rowley et al. 1998; Schneiderman 2000; Felzenszwalb et al. 2009). When the processing of individual image positions is expensive an exhaustive search over all positions can be computationally very demanding or even infeasible. This problem can, however, be mitigated with a fast and lightweight mechanism to reduce the number of candidate positions for which the more expensive computations are being performed (see e.g. Lampert and Blaschko 2008; Harzallah et al. 2009; Alexe et al. 2010).

We believe that by further increasing the number of layers in the model in combination with appropriate constraints on the connectivity we will be able to make progress with respect to both of these questions. As demonstrated in Sect. 5.2.2 the hierarchical formulation in combination with joint training leads to a 'separation of concerns' across layers, in which the lower layer is responsible for the local details while the higher layer determines primarily the overall pose. This allows the model to *learn* some degree of small-scale invariances, achieving an effect similar to the pooling layers e.g. in Lee te al. (2009) (but without having to explicitly build them in). We expect that a deeper model, in which such effects will be replicated across several layers, will be able to handle larger invariances, and that it will also allow us to work with shapes at higher resolutions while avoiding overfitting.

The third question is how to handle real-world images that contain not just one but many objects. This will make it necessary to model the interactions between the shapes of multiple occluding objects. Although the multi-part SBM can model multiple regions it is unlikely to be a good model of the regions that are the result of occlusion, as discussed in Le Roux et al. (2011). Their proposed solution is, in principle, directly applicable to the SBM and we are currently investigating how their or similar approaches can be utilized.

7 Conclusions

In this paper we have presented the Shape Boltzmann Machine, a strong generative model of object shape. The SBM is based on the general DBM architecture, a form of undirected graphical model that makes heavy use of latent variables to model high-order dependencies between the observed variables. We believe that the *combination* of (a) carefully chosen connectivity and capacity constraints, along with (b) a hierarchical architecture, and (c) a training procedure that allows for the joint optimization of the full model, is key to the success of the SBM.

These ingredients allow the SBM to learn high quality probability distributions over object shapes from small datasets, consisting of just a few hundred training images. The learned models are convincing in terms of both realism of samples from the distribution and generalization to new examples of the same shape class. Without making use of specialist knowledge about the shapes the model develops a natural representation with some separation of concerns across layers.

Overall we believe that by integrating powerful component models like the SBM into comprehensive generative models of images, performance in many computer vision tasks can be improved. We believe this to be a very promising direction of research.

Acknowledgments The majority of this work was performed whilst AE and NH were at Microsoft Research in Cambridge. Thanks to Charless Fowlkes and Vittorio Ferrari for access to datasets, and to Pushmeet Kohli for valuable discussions. AE acknowledges funding from the Carnegie Trust, the SORSAS scheme, and the IST Programme of the European Community under the PASCAL2 Network of Excellence (IST-2007-216886). NH acknowledges funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant agreement no. 270327, and from the Gatsby Charitable foundation. We finally thank the anonymous referees for their comments which helped improve the paper.

References

Ackley, D., Hinton, G., & Sejnowski, T. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, *9*(1), 147–169.

Alexe, B., Deselaers, T., & Ferrari, V. (2010a). ClassCut for unsupervised class segmentation. In *European Conference on Computer vision* (pp. 380–393).

Alexe, B., Deselaers, T., & Ferrari, V. (2010b). What is an object?. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 73–80).

Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., & Davis, J. (2005). SCAPE: Shape completion and animation of people. *ACM Transactions on Graphics (SIGGRAPH)*, *24*(3), 408–416.

Bertozzi, A., Esedoglu, S., & Gillette, A. (2007). Inpainting of binary images using the Cahn–Hilliard equation. *IEEE Transactions on Image Processing*, *16*(1), 285–291.

Bo, Y., & Fowlkes, C. (2011). Shape-based pedestrian parsing. In *IEEE Conference on Computer Vision and Pattern Recognition 2011*.

Borenstein, E., Sharon, E., & Ullman, S. (2004). Combining top-down and bottom-up segmentation. In *CVPR Workshop on Perceptual Organization in Computer Vision*.

Boykov, Y., & Jolly, M. P. (2001). Interactive graph cuts for oOptimal boundary & region segmentation of objects in N-D images. In *International Conference on Computer Vision 2001* (pp. 105–112).

Bridle, J. S. (1990). Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In *Advances in Neural Information Processing Systems* (Vol. 2, pp. 211–217).

Cemgil, T., Zajdel, W., & Krose, B. (2005). A hybrid graphical model for robust feature extraction from video. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1158–1165).

Chan, T. F., & Shen, J. (2001). Nontexture inpainting by curvature-driven diffusions. *Journal of Visual Communication and Image Representation*, *12*(4), 436–449.

Chen, F., Yu, H., Hu, R., & Zeng, X. (2013). Deep learning shape priors for object segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1870–1877).

Cootes, T., Taylor, C., Cooper, D. H., & Graham, J. (1995). Active shape models—Their training and application. *Computer Vision and Image Understanding*, *61*, 38–59.

Desjardins, G., & Bengio, Y. (2008). *Empirical evaluation of convolutional RBMs for vision*. Tech. Rep. 1327, Département d'Informatique et de Recherche Opérationnelle, Université de Montréal.

Eslami, S. M. A., & Williams, C. K. I. (2011). Factored shapes and appearances for parts-based object understanding. In *British Machine Vision Conference 2011*, (pp. 18.1–18.12).

Eslami, S. M. A., & Williams, C. K. I. (2012). A generative model for parts-based object segmentation. In P. Bartlett, F. Pereira, C. Burges, L. Bottou, & K. Weinberger (Eds.), *Advances in Neural Information Processing Systems* (Vol. 25, pp. 100–107). Red Hook, NY: Curran Associates, Inc.

Fei-Fei, L., Fergus, R., Perona, P. (2004). Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. In *IEEE Conference on Computer Vision and Pattern Recognition 2004, Workshop on Generative-Model Based Vision*.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2009). Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *99*, 1–19.

Freund, Y., & Haussler, D. (1994). *Unsupervised learning of distributions on binary vectors using two layer networks*, Tech. Rep. UCSC-CRL-94-25. Santa Cruz: University of California.

Frey, B., Jojic, N., & Kannan, A. (2003). Learning appearance and transparency manifolds of occluded objects in layer. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 45–52).

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*(1), 1–22.

Gavrila, D. M. (2007). A Bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*, 1408–1421.

Harzallah, H., Jurie, F., & Schmid, C. (2009). Combining efficient object localization and image classification. In *International Conference on Computer Vision*.

Heess, N., Roux, N. L., & Winn, J. M. (2011). Weakly supervised learning of foreground-background segmentation using masked RBMs. In *International Conference on Artificial Neural Networks* (Vol. 2, pp. 9–16).

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

Jojic, N., & Caspi, Y. (2004). Capturing image structure with probabilistic index maps. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 212–219).

Jojic, N., Perina, A., Cristani, M., Murino, V., & Frey, B. (2009). Stel component analysis: Modeling spatial correlations in image class structure. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2044–2051).

Kapoor, A. & Winn, J. (2006). Located hidden random fields: Learning discriminative parts for object detection. In *European Conference on Computer Vision* (pp. 302–315).

Kohli, P., Kumar, M. P., Torr, P. H. S. (2007). P3 & beyond: Solving energies with higher order cliques. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Kohli, P., Ladicky, L., & Torr, P. H. S. (2009). Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, *82*(3), 302–324.

Komodakis, N. & Paragios, N. (2009). Beyond pairwise energies: Efficient optimization for higher-order mrfs. In *IEEE Conference on Computer Vision and Pattern Recognition 2007* (pp. 2985–2992).

Kumar, P., Torr, P., & Zisserman, A. (2005). OBJ CUT. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 18–25).

Lampert, C. H., Blaschko, M., & Hofmann, T. (2008). Beyond sliding windows: Object localization by efficient subwindow search. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1–8).

Le Roux, N., & Bengio, Y. (2008). Representational power of restricted Boltzmann machines and deep belief networks. *Neural Computation*, *20*(6), 1631–1649.

Le Roux, N., Heess, N., Shotton, J., & Winn, J. (2011). Learning a generative model of images by factoring appearance and shape. *Neural Computation*, *23*(3), 593–650.

Lee, H., Grosse, R., Ranganath, R., & Ng, A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of Hierarchical representations. In *International Conference on Machine Learning* (pp. 609–616).

Morris, R. D., Descombes, X., & Zerubia, J. (1996). The Ising/Potts model is not well suited to segmentation tasks. In *Proceedings of the IEEE Digital Signal Processing Workshop*.

Murray, I., & Salakhutdinov, R. (2009). Evaluating probabilities under high-dimensional latent variable models. In *Advances in Neural Information Processing Systems* (Vol. 21).

Neal, R. M. (1992). Connectionist learning of belief networks. *Artificial Intelligence*, *56*, 71–113.

Neal, R. M. (2001). Annealed importance sampling. *Statistics and Computing*, *11*(2), 125–139.

Norouzi, M., Ranjbar, M., & Mori, G. (2009). Stacks of convolutional restricted Boltzmann machines for shift-invariant feature learning. In *CVPR* (pp. 2735–2742).

Nowozin, S., & Lampert, C. H. (2009). Global connectivity potentials for random field models. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 818–825).

Raina, R., Madhavan, A., & Ng, A. Y. (2009). Large-scale deep unsupervised learning using graphics processors. In *International Conference on Machine Learning* (pp. 873–880).

Ranzato, M., Mnih, V., & Hinton, G. E. (2010). How to generate realistic images using gated MRFs. In J. Lafferty, C. K. I. Williams, R. Zemel, J. Shawe-Taylor, & A. Culotta (Eds.), *Advances in Neural Information Processing Systems* (Vol. 23). Cambridge: MIT Press.

Ranzato, M., Susskind, J., Mnih, V., & Hinton, G. E. (2011). On deep generative models with applications to recognition. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2857–2864).

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, *22*(3), 400–407.

Roth, S., & Black, M. J. (2005). Fields of experts: A framework for learning image priors. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 860–867).

Rother, C., Kolmogorov, V., & Blake, A. (2004). "GrabCut": Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (SIGGRAPH)*, *23*, 309–314.

Rother, C., Kohli, P., Feng, W., & Jia, J. (2009). Minimizing sparse higher order energy functions of discrete variables. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1382–1389).

Rowley, H., Baluja, S., & Kanade, T. (1998). Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(1), 23–38.

Russell, B., Torralba, A., Murphy, K., & Freeman, W. (2008). LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision*, *77*, 157–173.

Salakhutdinov, R. & Hinton, G. (2009). Deep Boltzmann machines. In *International Conference on Artificial Intelligence and Statistics 2009*, (Vol. 5, pp. 448–455).

Salakhutdinov, R., & Murray, I. (2008). On the quantitative analysis of deep belief networks. In *International Conference on Machine Learning 2008*.

Schneiderman, H. (2000). *A statistical approach to 3D object detection applied to faces and cars*. PhD Thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

Shekhovtsov, A., Kohli, P., & Rother, C. (2012). Curvature prior for MRF-based segmentation and shape inpainting. In *DAGM/OAGM Symposium* (pp. 41–51).

Sigal, L., Balan, A., & Black, M. (2010). HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision*, *87*(1–2), 4–27.

Thomas, A., Ferrari, V., Leibe, B., Tuytelaars, T., & Gool, L. V. (2009). Using multi-view recognition and meta-data annotation to guide a robot's attention. *International Journal of Robotics Research*, *28*(8), 976–998.

Tieleman, T. (2008). Training restricted Boltzmann machines using approximations to the likelihood gradient. In *International Conference on Machine Learning 2008* (pp. 1064–1071).

Tjelmeland, H., & Besag, J. (1998). Markov random fields with higher-order interactions. *Scandinavian Journal of Statistics*, *25*(3), 415–433.

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

Winn, J., & Jojic, N. (2005). LOCUS: Learning object classes with unsupervised segmentation. In *International Conference on Computer Vision* (pp. 756–763).

Younes, L. (1999). On the convergence of Markovian stochastic algorithms with rapidly decreasing ergodicity rates. In *Stochastics and Stochastics Reports* (Vol. 65, pp. 177–228).

Younes, L., & Sud, P. (1989). Parametric inference for imperfectly observed Gibbsian fields. *Probability Theory and Related Fields*, *82*, 625–645.