DISJUNCTIVE NORMAL SHAPE BOLTZMANN MACHINE

Ertunc Erdil[⋆] *Fitsum Mesadi*[†] *Tolga Tasdizen*[†] *Mujdat Cetin*[⋆]

[⋆] Faculty of Engineering and Natural Sciences, Sabanci University, Tuzla, Istanbul, Turkey
{ertuncerdil, mcetin}@sabanciuniv.edu
[†] Department of Electrical and Computer Engineering University of Utah, Salt Lake City, UT, USA
u0883644@utah.edu, tolga@sci.utah.edu

ABSTRACT

Shape Boltzmann machine (a type of Deep Boltzmann machine) is a powerful tool for shape modelling; however, has some drawbacks in representation of local shape parts. Disjunctive Normal Shape Model (DNSM) is a strong shape model that can effectively represent local parts of objects. In this paper, we propose a new shape model based on Shape Boltzmann Machine and Disjunctive Normal Shape Model which we call Disjunctive Normal Shape Boltzmann Machine (DNSBM). DNSBM learns binary distributions of shapes by taking both local and global shape constraints into account using a type of Deep Boltzmann Machine. The samples generated using DNSBM look realistic. Moreover, DNSBM is capable of generating novel samples that differ from training examples by exploiting the local shape representation capability of DNSM. We demonstrate the performance of DNSBM for shape completion on two different data sets in which exploitation of local shape parts is important for capturing the statistical variability of the underlying shape distributions. Experimental results show that DNSBM is a strong model for representing shapes that are composed of local parts.

***Index Terms*—** Shape Boltzmann Machine, Disjunctive Normal Shape Model, Shape Sampling, Gibbs Sampling

1. INTRODUCTION

Shape modelling has a variety of applications in computer vision and image processing including object detection and image segmentation [1] [2] [3] [4], shape matching [5], inpainting [6], and graphics [7] [8]. In general, using a better shape model in these applications leads to better performance.

A strong shape model should contain two important properties: realism and generalization [9]. The first property states that the model should capture the correct shape distributions, i.e., samples that are drawn from the distribution should be valid shapes. The second constraint ensures that the samples generated from the learned distribution should also cover unseen but valid shapes. There exist a variety of approaches for $2D$ shape modelling in the literature [10] [11] [12] [13] [14]. Shape Boltzmann machine (SBM) [9] is a type of Deep Boltzmann machine (DBM) [15] designed for binary shape modelling. SBM learns binary distributions from a set of binary training shapes and generates samples from the learned distribution using block-Gibbs sampling. The advantage of SBM over other undirected shape models (Restricted Boltzmann Machine (RBM) [16] and DBM [15]) is its ability to learn shape distributions when the training set is limited. Local shape representation of SBM enables the model to generate novel samples by exploiting local shape parts when generating a new sample. It divides a given shape into four slightly overlapping equal-sized patches as shown with different colors in the first row of Fig. 1, where each patch plays the role of a local shape part. However, these patches do not correspond to a geometrically meaningful local shape parts. Here, a geometrically meaningful local shape part stands for a single physical component of the shape, for example, a particular limb (e.g., head, arm, etc.) of the standing person shown in Fig. 1. In patch-based local shape representation, a geometrically meaningful local shape part can appear in multiple patches. For example, the left arm of the standing person shown in the first row of Fig. 1 is contained partially in both red and yellow local regions in the first training image. Therefore, samples generated by SBM may contain unrealistic samples. For example, the sample in the third column of the first row in Fig. 1 contains two left arms; one is raised up and the other partially appears just to the left of the body.

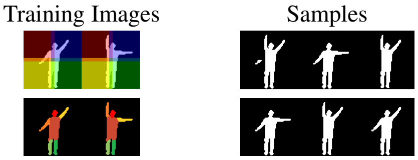
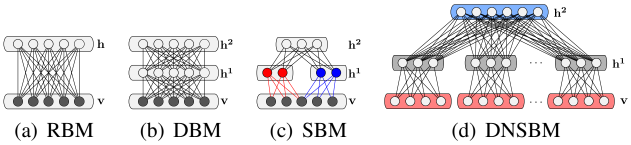


Fig. 1: Local shape representation and shape sampling using SBM (first row) and the proposed DNSBM (second row).

Our contribution in this paper is a new shape model called Disjunctive Normal Shape Boltzmann Machine (DNSBM) which exploits the property of SBM for learning complex binary distributions and the property of DNSM [1] for representing local parts of shapes. DNSM is an implicit and parametric model that represents a shape by a union of convex polytopes. In DNSM, each polytope or union of a subset of the polytopes can represent a physical local part of an object as shown in the second row of Fig. 1. This property of DNSM makes it a very powerful model for representing local shape parts. As we exploit that property, samples generated by our proposed DNSBM are realistic. Also, DNSBM is able to generate novel samples which are not contained in the training set by exploiting local shape parts in block-Gibbs sampling and by using the learned distribution. We train DNSBM on two different data sets in which local shape parts are important for capturing the statistical variability of the whole shape distribution and show its performance by generating samples from the distribution for shape completion. Experimental results show the effectiveness of DNSBM. Some exemplary results of DNSBM using two training examples are shown in the second row of Fig. 1. Here, our approach is able to generate realistic and novel samples that are not contained in the training set.

This work has been supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under Grant 113E603.



(a) RBM (b) DBM (c) SBM (d) DNSBM

Fig. 2: Undirected models for modelling binary shapes.

2. RELATED WORK

Restricted Boltzmann Machine (RBM) [16] is a model that includes a number of hidden variables $\mathbf{h}$ each connected to all image pixels (units in the visible layer $\mathbf{v}$) as shown in Fig. 2(a). Note that there are no direct connections between the units of a layer, which makes this a bipartite graph. Hence, the energy of a configuration can be written as follows:

$$E(\mathbf{v},\mathbf{h}) = \sum_i b_i v_i + \sum_{i,j} w_{ij} v_i h_j + \sum_j c_j h_j \tag{1}$$

where, i and j range over pixels and hidden variables, respectively. Then, the model can learn constraints and dependencies between pixels by learning the parameters w_{ij}, b_i, and c_j. The distribution over $\mathbf{v}$ is given by marginalizing over the hidden variables: $p(\mathbf{v}) = \sum_{\mathbf{h}} \exp(-E(\mathbf{v},\mathbf{h}))/Z(\theta)$, where θ represents the model parameters and $Z(\theta)$ is the partition function. This marginalization allows the model to capture dependencies between the image pixels. RBM has edges between hidden and visible variables. Therefore, all hidden units are conditionally independent given the visible units. Similarly, all visible units are conditionally independent given the hidden units. This property is useful for exact and efficient inference. Then, the conditional probabilities can be written as $p(v_i = 1|\mathbf{h}) = \sigma(\sum_j w_{ij} h_j + b_i)$ and $p(h_j = 1|\mathbf{v}) = \sigma(\sum_i w_{ij} v_i + c_j)$ where, $\sigma(\circ) = 1/(1 + \exp(-\circ))$ is the sigmoid function. Using this property, $\mathbf{v}$ and $\mathbf{h}$ can be sampled consecutively, which can be exploited during learning the model parameters [17].

RBMs can approximate any binary distribution if an exponential number of hidden units and a large amount of training data are available [16]. The DBM is capable of learning more complex structures in the data using additional hidden units as shown in Fig. 2(b). The energy of a DBM with two hidden layers can be written as follows:

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

where, i, j, and k range over pixels, the first, and the second hidden variables, respectively. Exact inference is no longer possible in this model, however, the conditional distributions $p(\mathbf{v}|\mathbf{h^1})$, $p(\mathbf{h^1}|\mathbf{v},\mathbf{h^2})$ and $p(\mathbf{h^2}|\mathbf{h^1})$ can be computed as in RBMs [15]. Then, computationally efficient approximate inference can be established by block-Gibbs sampling from the posterior $p(\mathbf{h^1},\mathbf{h^2}|\mathbf{v})$ [9].

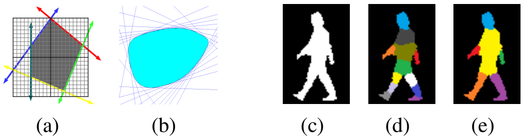
RBM and DBM are powerful models, however, they require a large number of binary images to learn the shape distributions like the other recent and powerful generative models: Generative Adversarial Network (GAN) [18] and Variational Autoencoders (VAE) [19]. In most applications, sizes of the available data sets are small since obtaining segmented binary images is an expensive process. SBM [9] is a shape model based on RBM and DBM that accurately captures the properties of binary shapes. Unlike RBM and DBM, SBM is capable of learning shape distributions even when the size of the training set is limited, by exploiting information from local shape representations. The visible units $\mathbf{v}$ of the SBM are the pixels of an $X \times Y$ binary image. SBM divides images into four equal-sized slightly overlapping patches to represent local shape parts as shown in Fig. 1. The first hidden layer $\mathbf{h^1}$ consists of four blocks and each block is fully connected to a particular patch. Finally, all units in $\mathbf{h^1}$ are fully connected to the units in the second hidden layer $\mathbf{h^2}$. The structure of SBM for $1D$ images is shown in Fig. 2(c). The structure can easily be generalized to $2D$. SBM follows the procedure in [15] to learn the model parameters and generates a new sample using block-Gibbs sampling.

Recently, Erdil et al. [20] proposed a Markov chain Monte Carlo method for generating samples from shape posterior densities. Since the method represents local shape parts with patches as in SBM, it suffers from similar issues when generating a new sample.

3. DISJUNCTIVE NORMAL SHAPE BOLTZMANN MACHINE

3.1. Binary shape representation using DNSM

DNSM represents a shape by a union of convex polytopes. A polytope can be represented by intersection of half-spaces as shown in Fig. 3(a). Smooth convex polyopes can be obtained by increasing number of half-spaces (see Fig. 3(b)).



(a) (b) (c) (d) (e)

Fig. 3: DNSM shape representation.

DNSM approximates the characteristic function of a shape as a union of convex polytopes which themselves are represented as intersections of half-spaces. Consider the characteristic function of a D-dimensional shape $f : \mathbf{R}^D \rightarrow B$ where $B = \{0, 1\}$. Let $\Omega^+ = \{\mathbf{x} \in \mathbf{R}^D : f(\mathbf{x}) = 1\}$ represent the foreground region. Ω^+ can be approximated as a union of N convex polytopes, $\Omega^+ \approx \bigcup_{i=1}^N P_i$. The i^{th} polytope is defined as the intersection $P_i = \bigcap_{j=1}^{M_i} H_{ij}$ of M_i half-spaces. The half-spaces are defined as $H_{ij} = \{\mathbf{x} \in \mathbf{R}^D : h_{ij}(\mathbf{x})\}$ where

$$h_{ij}(\mathbf{x}) = \begin{cases} 1, & \text{if} \sum_{k=1}^D \delta_{ijk} x_k + c_{ij} \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

Therefore, Ω^+ is approximated by $\bigcup_{i=1}^N \bigcap_{j=1}^{M_i} H_{ij}$ and equivalently $f(\mathbf{x})$ is approximated by the disjunctive normal form $\bigvee_{i=1}^N \bigwedge_{j=1}^{M_i} h_{ij}(\mathbf{x})$. Converting the disjunctive normal form to a differentiable shape representation requires the following steps: First, De Morgan's rules are used to replace the disjunction with negations and conjunctions, which yields $f(\mathbf{x}) \approx \bigvee_{i=1}^N \bigwedge_{j=1}^{M_i} h_{ij}(\mathbf{x}) = \neg \bigwedge_{i=1}^N \neg \bigwedge_{j=1}^{M_i} h_{ij}(\mathbf{x})$. Since conjunctions of binary functions are equivalent to their product and negation is equivalent to subtraction from 1, $f(\mathbf{x})$ can also be approximated as $1 - \prod_{i=1}^N (1 - \prod_{j=1}^{M_i} h_{ij}(\mathbf{x}))$. The final step for obtaining a differentiable representation is to relax the discriminants h_{ij} to sigmoid functions $\sigma_{ij} = 1/(1 + e^{-(\sum_{k=1}^D \delta_{ijk} x_k + c_{ij})})$. The resulting approximation to the shape characteristic functions is then given by $f(\mathbf{x}) = 1 - \prod_{i=1}^N \left(1 - \prod_{j=1}^{M_i} \sigma_{ij}\right)$, where $\mathbf{x} = \{x, y\}$ for two-dimensional ($2D$) shapes and $\mathbf{x} = \{x, y, z\}$ for three-dimensional ($3D$) shapes [1].

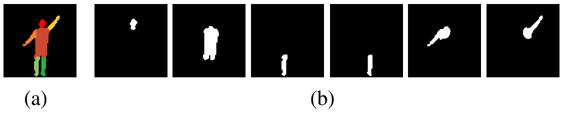

(a) (b)

Fig. 4: Decomposing a shape into polytopes. (a) A shape with DNSM representation. (b) Binary images corresponding to each physical shape part (polytope).

The only free parameters of the model are δ_{ijk} and c_{ij}, which determine the orientation and location of the sigmoid functions (discriminants) that define the half-spaces. The level set $f(x) = 0.5$ is taken to represent the interface between the foreground ($f(\mathbf{x}) \geq 0.5$) and background ($f(\mathbf{x}) < 0.5$) regions.

The DNSM discriminant parameters, $\mathbf{\Delta}^t$, that represent the t^{th} training sample can be obtained by choosing the weights that minimize the following energy function

$$E(\mathbf{\Delta}^t) = \int_{\mathbf{x}\in\Omega} (f(\mathbf{x}) - q_t(\mathbf{x}))^2 d\mathbf{x} + \eta \sum_{i}^{N} \sum_{r\neq i}^{N} \int_{\mathbf{x}\in\Omega} g_i(\mathbf{x}) g_r(\mathbf{x}) d\mathbf{x} \quad (3)$$

where, $g_i(\mathbf{x}) = \prod_{j=1}^{M_i} \sigma_{ij}$ represents the individual polytopes of $f(\mathbf{x})$. $q_t(\mathbf{x})$ is the t^{th} binary training image (1 for object and 0 for background) to be represented by DNSM and η is a constant that controls the allowed degree of overlap between polytopes. We find that having slightly overlapping polytopes is important to ensure shape continuity in the generated samples by DNSBM. We minimize Equation (3) using gradient descent to obtain $\mathbf{\Delta}^t$ which represents the t^{th} training sample. DNSM representation of the binary image in Fig. 3(c) is given in Fig. 3(d). Note that each polytope may not correspond to a local geometrically meaningful shape part since large number of convex polytopes are required for representing complex shapes. One can consider combining polytopes manually to obtain local shape parts when constructing the training set. We use the approach proposed in [21] that relaxes the convexity constraint of DNSM and represents complex shapes by a smaller number of approximately convex polytopes each corresponding to a geometrically meaningful local shape part. Fig. 3(e) shows the approximately convex polytopes obtained using the approach in [21].

3.2. From DNSM to DNSBM

Our proposed approach, DNSBM is a type of Deep Boltzmann Machine having the structure shown in Fig. 2(d). In DNSBM, each pre-aligned binary training shape in an $X \times Y$ image is initially represented with N polytopes such that each polytope corresponds to a physically meaningful (local) shape part as explained in Section 3.1. Then, each shape is decomposed into N binary images where each binary image represents a single local shape part as shown in Fig. 4. Each red block in the visible layer $\mathbf{v}$ of DNSBM (see Fig. 2(d)) corresponds to a binary image that represents a particular local shape part. Therefore, there are N red blocks each containing $X \times Y$ units in the visible layer of DNSBM as exemplified by the binary images in Fig. 4(b). The first hidden layer $\mathbf{h^1}$ of DNSBM is composed of N blocks (shown in gray in Fig. 2(d)). The units in each block of $\mathbf{v}$ are fully connected with the units in the corresponding block of $\mathbf{h^1}$. Each unit of $\mathbf{h^1}$ is also connected to all units of $\mathbf{h^2}$. While the connections between $\mathbf{v}$ and $\mathbf{h^1}$ capture the dependencies between pixels, the connections between $\mathbf{h^1}$ and $\mathbf{h^2}$ capture the inter-relations of local shape parts.

Learning of the model involves maximizing $\log p(\mathbf{v}; \theta)$ of the observed data $\mathbf{v}$ with respect to its parameters $\theta = \{b, W_1, W_2, c_1, c_2\}$. The work in [15] proposes a two-phase learning procedure. In the pre-training, the model is trained bottom up, one layer at a time, to find a good initial estimates of the model parameters. Once the parameters are initialized, parameters of the full model can be fine-tuned by backpropagation. In DNSBM, each connected red-gray block pair between $\mathbf{v}$ - $\mathbf{h^1}$ and each connected gray-blue pair between $\mathbf{h^1}$ - $\mathbf{h^2}$ forms an RBM. Although a more effective learning of the model parameters using the procedure in [15] is possible, we found sufficient to train each RBM in DNSBM from bottom-up in a greedy manner using approximate gradient descent [22]. Once the parameters of DNSBM are found, we generate samples from the model using block-Gibbs sampling.

4. EXPERIMENTAL RESULTS

In this section, we present experimental results of DNSBM on two data sets in which local shape parts play an important role for identifying shape densities when the training set is limited. We compare the performance of the DNSBM with SBM. The implementation of DNSBM and the data sets are available at `github.com/eerdil/dnsbm_icassp17`.

The first data set is the standing person data set [23]. The data set contains 50, 170×170 binary images of a standing person with varying arm postures. We construct a training set with 28 images by using shapes each containing a particular posture of either left or right arm as shown in Fig. 5. Each of the remaining 22 shapes in the data set contains arm postures of both left and right arms. Since each arm posture is contained for both left and right arms separately in the training set, the remaining 22 shapes can be explored by exploiting these local shape parts. Note that, exploitation of local shape parts is not done simply by combining all possible local shapes, it naturally emerges as a result of block-Gibbs sampling. We obtain local shape (head, left arm, right arm, etc.) representations of the standing person for each binary training shape using DNSM. When training DNSBM on this data set, we empirically set sizes of $\mathbf{h^1}$ and $\mathbf{h^2}$ to 2000 and 500, respectively. Increasing the size of $\mathbf{h^2}$ may cause overfitting whereas $\mathbf{h^1}$ should be large enough to capture pixel dependencies.

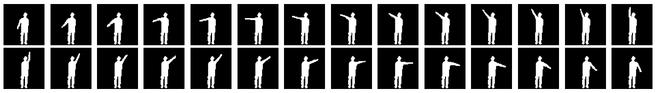


Fig. 5: Training set of the standing person data set.

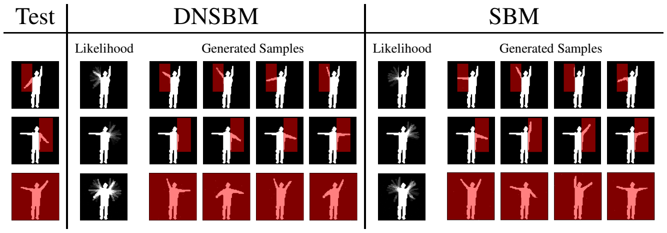


Fig. 6: Samples generated by DNSBM and SBM for completion of the shapes in the first column. Pixels in the red region are missing.

We design 3 test cases having different missing regions to be completed in our experiments as shown in the first column of Fig. 6. Image completion is established by generating samples from both DNSBM and SBM using the observed part of the shape. Some shape completion results of each approach are shown in Fig. 6. We also provide likelihood images in the first column for each approach in Fig. 6. These images are obtained by summing up all generated samples and normalizing with the total number of samples [24]. We further enhance the likelihood images in Fig. 6 for visualization purposes. Note that in the likelihood images, bright pixels indicate high occurrence of the corresponding pixel in foreground region of the generated samples. In this data set, all samples of DNSBM appear realistic, i.e., there is no sample that does not look like a standing person, whereas SBM generates some unrealistic samples (see the standing person samples in Fig. 9(b)).

The second data set is the walking silhouette data set [4]. The walking silhouette data set contains 150 binary images of a walking person. Similar to the experiments on the standing person data set, we choose a subset of 24 images (see Fig. 7) for training. We obtain the local shape parts of walking silhouettes using 6 polytopes with DNSM. We train the DNSBM on this data set using 1000 units for $\mathbf{h^1}$ and 50 units for $\mathbf{h^2}$ for 78 × 52 images.

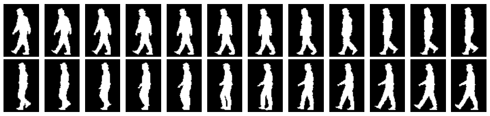


Fig. 7: Training set of the walking silhouette data set.

We design 5 test cases for shape completion using shapes not included in the training set and with different missing regions to be completed as shown in the first column of Fig. 8. We perform shape completion on these test images by generating samples from both DNSBM and SBM. Some completion results of each method together with the likelihood images for the corresponding input shape are shown in Fig. 8. The walking silhouette data set is a more challenging data set than the previous one since it contains more local shape parts that change their posture. In this data set, DNSBM produces better results than the SBM in terms of the number of realistic samples, as well as its generalization capability to generate valid and diverse shapes, as shown particularly in the 2^{nd}, 3^{rd}, and 5^{th} rows of Fig. 8. Some unrealistic samples generated by both DNSBM and SBM on the walking silhouette data set are given in Fig. 9. The patch-based local shape representation of SBM is not a good representation for this data set, since almost each physical shape part, especially legs of the silhouette, appears in more than one patch. This leads SBM to generate a large number of unrealistic samples in this data set.

Table 1: Comparison of DNSBM and SBM using Dice score.

	DNSBM	SBM
Walking silhouette	**0.6526**	0.6112
Standing Person	**0.5935**	0.5825

Quantitative evaluation of sampling-based approaches is not a trivial task and requires considering different metrics. First, we compute the similarity between the ground truth and the completion results using Dice score [25], since it is expected that a sampling-based approach generates many samples that are similar to the ground truth. The average Dice score results of all test cases for both data sets are shown in Table 1. Note that, high values of Dice score indicate higher similarity with the ground truth. Second, we expect to obtain realistic samples. We measure this by computing the probability of sampling the completed region given the observed data using the imputation score [9]. The average of all imputation scores in all test cases of both data sets are 0.085 for DNSBM and 0.014 for SBM where higher is better. Finally, a good sampling approach is expected to generate diverse samples. We demonstrate the diversity of samples by plotting the precision-recall (PR) values of all samples generated in all test cases in the walking silhouette data set as shown in Fig. 10. The results demonstrate that the samples of DNSBM spread in the precision-recall space more than the samples of SBM. Note that a large number of blue crosses in Fig. 10 correspond to unrealistic samples produced by SBM. Therefore, the superiority of the DNSBM over SBM in terms of diversity becomes more evident if we consider Fig. 10 without such samples.

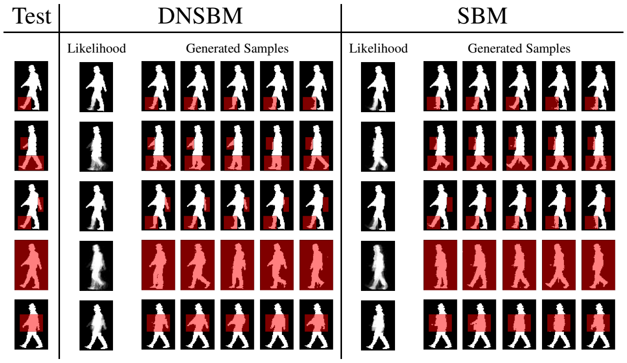
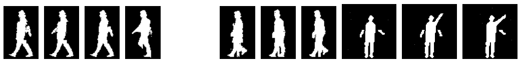


Fig. 8: Samples generated by DNSBM and SBM for completion of the shapes in the first column. Pixels in the red region are missing.



(a) DNSBM (b) SBM

Fig. 9: Some unrealistic samples generated by DNSBM and SBM.

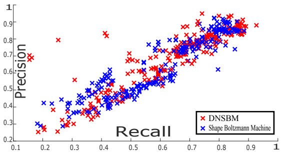


Fig. 10: PR values of the samples generated using the walking silhouette data set.

Since DNSBM uses a representation of each physical local part individually by a single polytope, it does not suffer from having multiple pieces for a single local part in the generated samples. However, in some cases, exploiting different local shape parts in the training set does not yield a visually appealing sample as shown in Fig. 9. This problem originates at places where local shape parts are connected to each other. Although we have solved this problem up to some level by generating overlapping polytopes, we can still encounter such samples in some rare cases. Some possible solutions of this problem might be incorporating information about tie locations of polytopes to the sampling process. One can also consider performing a local registration as a post-processing step.

5. CONCLUSION

We have presented a shape model, DNSBM, that is based on the SBM and the DNSM. DNSBM is able to represent physically meaningful local shape parts individually and exploits this representation when the training set size is limited. We have shown the performance of DNSBM on two data sets for shape completion. The proposed method exhibits better performance than SBM.

6. REFERENCES

[1] Fitsum Mesadi, Mujdat Cetin, and Tolga Tasdizen, "Disjunctive normal shape and appearance priors with applications to image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 703–710. Springer, 2015.

[2] John Winn and Nebojsa Jojic, "Locus: Learning object classes with unsupervised segmentation," in *IEEE International Conference on Computer Vision*. IEEE, 2005, vol. 1, pp. 756–763.

[3] Junmo Kim, Müjdat Çetin, and Alan S Willsky, "Nonparametric shape priors for active contour-based image segmentation," *Signal Processing*, vol. 87, no. 12, pp. 3021–3044, 2007.

[4] Daniel Cremers, Stanley J Osher, and Stefano Soatto, "Kernel density estimation and intrinsic alignment for shape priors in level set segmentation," *International Journal of Computer Vision*, vol. 69, no. 3, pp. 335–351, 2006.

[5] Dariu M Gavrila, "A Bayesian, exemplar-based approach to hierarchical shape matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 8, pp. 1408–1421, 2007.

[6] Tony F Chan and Jianhong Shen, "Nontexture inpainting by curvature-driven diffusions," *Journal of Visual Communication and Image Representation*, vol. 12, no. 4, pp. 436–449, 2001.

[7] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis, "Scape: shape completion and animation of people," in *ACM Transactions on Graphics*. ACM, 2005, vol. 24, pp. 408–416.

[8] James F Blinn, "A generalization of algebraic surface drawing," *ACM Transactions on Graphics*, vol. 1, no. 3, pp. 235–256, 1982.

[9] SM Ali Eslami, Nicolas Heess, Christopher KI Williams, and John Winn, "The shape Boltzmann machine: a strong model of object shape," *International Journal of Computer Vision*, vol. 107, no. 2, pp. 155–176, 2014.

[10] Yuri Y Boykov and Marie-Pierre Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *IEEE International Conference on Computer Vision*. IEEE, 2001, vol. 1, pp. 105–112.

[11] Sebastian Nowozin and Christoph H Lampert, "Global connectivity potentials for random field models," in *IEEE Conference Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 818–825.

[12] Carsten Rother, Pushmeet Kohli, Wei Feng, and Jiaya Jia, "Minimizing sparse higher order energy functions of discrete variables," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 1382–1389.

[13] Timothy F Cootes, Christopher J Taylor, David H Cooper, and Jim Graham, "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[14] Nisha Ramesh, Fitsum Mesadi, Mujdat Cetin, and Tolga Tasdizen, "Disjunctive normal shape models," in *IEEE International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2015, pp. 1535–1539.

[15] Ruslan Salakhutdinov and Geoffrey E Hinton, "Deep Boltzmann machines," in *International Conference on Artificial Intelligence and Statistics*, 2009, pp. 448–455.

[16] Yoav Freund and David Haussler, "Unsupervised learning of distributions on binary vectors using two layer networks," in *Advances in Neural Information Processing Systems*, 1992, pp. 912–919.

[17] Tijmen Tieleman, "Training restricted Boltzmann machines using approximations to the likelihood gradient," in *International Conference on Machine learning*. ACM, 2008, pp. 1064–1071.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[19] Diederik P Kingma and Max Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[20] Ertunc Erdil, Sinan Yildirim, Mujdat Cetin, and Tolga Tasdizen, "MCMC shape sampling for image segmentation with nonparametric shape priors," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 411–419.

[21] Fitsum Mesadi and Tolga Tasdizen, "Convex decomposition and efficient shape representation using deformable convex polytopes," *arXiv preprint arXiv:1606.07509*, 2016.

[22] Geoffrey E Hinton, "A practical guide to training restricted Boltzmann machines," in *Neural Networks: Tricks of the Trade*, pp. 599–619. Springer, 2012.

[23] Fei Chen, Huimin Yu, Roland Hu, and Xunxun Zeng, "Deep learning shape priors for object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1870–1877.

[24] Ayres C Fan, John W Fisher III, William M Wells III, James J Levitt, and Alan S Willsky, "MCMC curve sampling for image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 477–485. Springer, 2007.

[25] Lee R Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.