

Spatial Mixture Models with Learnable Deep Priors for Perceptual Grouping

Jinyang Yuan, Bin Li,* Xiangyang Xue

Shanghai Key Laboratory of Intelligent Information Processing
School of Computer Science, Fudan University
Fudan-Qiniu Joint Laboratory for Deep Learning
Shanghai Institute of Intelligent Electronics & Systems
{yuanjinyang, libin, xyxue}@fudan.edu.cn

Abstract

Humans perceive the seemingly chaotic world in a structured and compositional way with the prerequisite of being able to segregate conceptual entities from the complex visual scenes. The mechanism of grouping basic visual elements of scenes into conceptual entities is termed as perceptual grouping. In this work, we propose a new type of spatial mixture models with learnable priors for perceptual grouping. Different from existing methods, the proposed method disentangles the representation of an object into “shape” and “appearance” which are modeled separately by the mixture weights and the conditional probability distributions. More specifically, each object in the visual scene is modeled by one mixture component, whose mixture weights and the parameter of the conditional probability distribution are generated by two neural networks, respectively. The mixture weights focus on modeling spatial dependencies (i.e., shape) and the conditional probability distributions deal with intra-object variations (i.e., appearance). In addition, the background is separately modeled as a special component complementary to the foreground objects. Our extensive empirical tests on two perceptual grouping datasets demonstrate that the proposed method outperforms the state-of-the-art methods under most experimental configurations. The learned conceptual entities are generalizable to novel visual scenes and insensitive to the diversity of objects.

Introduction

The ability to perceive complex visual scenes in a structured and compositional way is crucial for humans to understand the seemingly chaotic world (Lake et al. 2017). Finding the underlying mechanisms of grouping basic visual elements is usually termed as the binding (Treisman 1996; Wolfe and Cave 1999), or more specifically, the perceptual grouping (Grossberg, Mingolla, and Ross 1997) problem, and has been studied extensively in the fields of neuroscience, cognitive science and psychology. Designing more human-like AI systems which learn compositional and disentangled representations (Bengio, Courville, and Vincent 2013) is desirable, because of their superior expressiveness and generalization ability compared to those systems learning a single complex representation. Complex visual scenes are composed of numerous relatively simpler primitive entities.

By hierarchically decomposing the observed visual scenes (Bienenstock, Geman, and Potter 1997), the raw observations can be summarized into organized and compact knowledge which is generalizable to an infinite number of novel scenes constructed by the combinations of these primitive entities (Biederman 1987; Hummel and Biederman 1992; van den Hengel et al. 2015).

Inspired by the synchronization theory (Milner 1974), various approaches have been proposed to tackle the perceptual grouping problem by using neuronal synchrony as the grouping mechanism (Wang and Terman 1995; Rao et al. 2008; Reichert and Serre 2014). Despite of the satisfactory grouping results already obtained, these methods do not learn separate representations for individual objects in the visual scene, which limits the expressiveness and generalization ability of the learned models. In order to learn disentangled and compositional representations of visual scenes, several methods which integrate spatial mixture models with neural networks have been proposed in recent years (Greff, Srivastava, and Schmidhuber 2015; Greff et al. 2016; Prémont-Schwarz et al. 2017; Greff, van Steenkiste, and Schmidhuber 2017). In these approaches, different objects are model by different mixture components, and each component is encouraged to focus on an individual object. The underlying intuition is that the image of visual scene can be decomposed into the pixel-wise weighted sum of reconstructed images of individual objects, with mixture weights computed based on similarities between the images of scene and objects at each pixel.

The perceptual grouping results of these recently proposed spatial mixture models are gratifying. Images of individual objects can be reconstructed accurately if objects do not overlap each other. However, there exist two potential downsides: 1) pixel-wise mixture weights are computed based on spatially dependent conditional probability distributions, and 2) the background is either not explicitly modeled by a mixture component or considered as an ordinary component. The first treatment leads to entangled learning of spatial dependencies of pixels because mixture weights shall vary accordingly as conditional probability distributions vary among pixels. The second treatment either complicates the determination of background pixels or increases the modeling complexity because the background is much harder to represent if considered as an ordinary

*Corresponding author

Copyright © 2019, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

component due to the diverse combinations of foreground objects.

In this paper, we propose a new type of spatial mixture models which places a learnable deep prior of spatial dependencies on mixture weights and considers modeling the background as a special component complementary to the foreground objects. The basic idea is that, in the proposed model, the mixture weights should focus on modeling the spatial dependencies (i.e., shapes) of the objects while the conditional probability distributions deal with variations (i.e., appearance) within objects. Instead of modeling mixture weights as coefficients with the only constraint that the sum of them equals one at each pixel, they are computed based on the outputs of a neural network with some hidden states as inputs. The inherent structure and learned parameters of the network act like priors of spatial dependencies and regularize the model to correctly estimate the shapes of objects in the visual scenes. By constructing the mixture weights in an innovative manner inspired by the truncated stick-breaking process, foreground objects and the background which is harder to represent are modeled differently. Because the background is considered as a special component whose mixture weights are implicitly determined by estimations of less varying foreground objects, the proposed method can thus achieve better compositionality and generalizes better to novel visual scenes.

We evaluate our method on two perceptual grouping datasets, in which images are composed of simple shapes or handwritten images, under different experimental configurations. Extensive empirical results suggest that representing the complex regions of background pixels in a compositional manner is crucial to high-quality grouping results. Compared with two state-of-the-art perceptual grouping algorithms, the proposed method not only achieves comparable or higher grouping accuracies, but also better estimates the images of individual objects from the image of visual scene. By disentangling the representation of an object into “shape” and “appearance” which are modeled separately by the mixture weights and the conditional probability distributions, the proposed method can better learn the spatial dependencies of pixels. The learned conceptual entities are generalizable to novel visual scenes and insensitive to the diversity of objects.

Preliminaries

Perceptual Grouping

Perceptual grouping is a type of binding problem that studies how brains arrange elements of visual scenes into conceptual entities like objects or patterns. Based on the inductive biases that are either innate or learned from experiences, humans tend to segregate, for example, the upper-left corner image in Figure 1a into three white hollow shapes and one black background although these shapes are connected and partially overlapped. The underlying mechanisms have been studied extensively for years. Gestalt psychologists suggested that humans perceive meaningful contents from the sensory information based on inborn mental laws, and summarized their theories into several rule-based principles

of perception (e.g. proximity, similarity, continuity, closure, and common fate) which is often known as Gestalt laws of grouping (Goldstein and Brockmole 2016). The principle of common fate differs from others in that it requires temporal information and is only applicable when relative motions exist. In this work, we only consider utilization of spatial information and focus on building models which follow the principles of proximity, similarity, continuity, and closure by learning prior knowledge of spatial dependencies from data in an unsupervised setting.

Spatial Mixture Model

In spatial mixture models, pixels are assumed to be generated independently from the mixture. Correlations between pixels can be achieved by assuming conditional probability distributions, or mixture weights, or both of them to be spatially dependent. Let X_m be the m th pixel of image \mathbf{X} , and Z_m be the latent variable specifying the mixture component of X_m . $\theta_{k,m} = \{\theta_{k,m}^c, \theta_{k,m}^w\}$ represents parameters for generating X_m in the k th component. For the sake of notational simplicity, we let $p_{m,k} = P(X_m|Z_m=k; \theta_{k,m}^c)$ and $\pi_{m,k} = P(Z_m=k; \theta_{:,m}^w) = \theta_{k,m}^w$. The log probability of observing image \mathbf{X} is given by

$$\log P(\mathbf{X}; \theta) = \sum_m \log \sum_k p_{m,k} \pi_{m,k}$$

Computing the maximum likelihood estimate (MLE) of θ directly is difficult due to the existence of unobserved component assignments. Approximated estimates can be obtained iteratively by applying the Expectation-Maximization (EM) algorithm. Let $\gamma_{m,k}^{(t)}$ denote the posterior probability at the t th iteration $P(Z_m=k|X_m; \theta_{k,m}^{(t)})$, which is computed by

$$\gamma_{m,k}^{(t)} = \frac{p_{m,k}^{(t)} \pi_{m,k}^{(t)}}{\sum_{k'} p_{m,k'}^{(t)} \pi_{m,k'}^{(t)}} \quad (1)$$

Parameters are updated using $\theta^{(t+1)} = \arg \max_{\theta} Q(\theta; \theta^{(t)})$, where $Q(\theta; \theta^{(t)})$ is the Q-function defined as

$$Q(\theta; \theta^{(t)}) = \sum_m \sum_k \gamma_{m,k}^{(t)} (\log p_{m,k} + \log \pi_{m,k}) \quad (2)$$

Neural Expectation Maximization

Neural Expectation Maximization (N-EM) (Greff, van Steenkiste, and Schmidhuber 2017) is a state-of-the-art spatial mixture model for solving the perceptual grouping problem. In N-EM, both conditional probability distributions and mixture weights are modeled as spatially dependent. Parameters $\theta_{k,m}^c = f_{\phi}(s_k)_m$ are generated by a neural network f_{ϕ} with some hidden states s_k as inputs, and mixture weights $\pi_{m,k}$ are computed based on $\theta_{:,m}^c$.

Because of the non-linearity of neural networks, the optimal hidden state s_k^* that maximizes $Q(\theta; \theta^{(t)})$ does not have a closed-form solution. Two types of generalized EM algorithms were proposed to solve this problem. One is to utilize the gradient descent to optimize s_k iteratively

$$s_k^{(t+1)} = s_k^{(t)} + \eta \sum_m \gamma_{m,k}^{(t)} \frac{\partial \log p_{m,k}}{\partial \theta_{k,m}^c} \frac{\partial \theta_{k,m}^c}{\partial s_k} \Big|_{s_k=s_k^{(t)}} \quad (3)$$

Another is to mimic the process of gradient descent by a recurrent neural network (RNN). The neural network f_ϕ is trained by the backpropagation through time (BPTT) algorithm (Robinson and Fallside 1987; Werbos 1988).

The loss function of each image is given by

$$L = - \sum_m \sum_k \gamma_{m,k} (\log p_{m,k} + \log \pi_{m,k}) + \lambda \sum_m \sum_k (1 - \gamma_{m,k}) D_{\text{KL}}(p_{\text{prior}} || p_{m,k}) \quad (4)$$

This loss function consists of two parts. The first part is the negative Q-function which measures the reconstruction errors. The second part is the weighted sum of the Kullback-Leibler (KL) divergences between the conditional probabilities $p_{m,k}$ and a distribution p_{prior} predefined by the prior knowledge of the image (e.g. the background intensities). $p_{m,k}$ is encouraged to be close to the distribution of background when the k th component is unconfident that the m th pixel should be assigned to it ($\gamma_{m,k}$ is small).

Spatial Mixture Model with Learnable Deep Priors

Except the implicit restriction that the mixture weights at each pixel should sum up to 1 ($\sum_k \pi_{m,k} = 1, \forall m$), N-EM does not place any constraint on $\pi_{m,k}$. Mixture weights are assumed to be spatially dependent in N-EM, and the optimal weights $\pi_{m,k}^*$ that maximizes the Q-function is given by $\pi_{m,k}^* = \gamma_{m,k} / \sum_{k'} \gamma_{m,k'} = \gamma_{m,k}$. The ability to distinguish objects with similar intensities relies on spatially dependent conditional probabilities with parameters generated by a neural network. One downside of this approach is the assumption of spatial dependencies for both conditional probabilities and mixture weights, which leads to entangled learning of correlations between pixels. This assumption also does not fit well with the principle of similarity (elements composing the same object should resemble each other) in the Gestalt laws of grouping. Moreover, the background is not explicitly modeled by a mixture component.

To better follow the principle of similarity and model the background pixels, we integrate neural networks and the EM algorithm in a different way. Instead of modeling correlations between pixels by generating spatially dependent parameters of conditional probabilities $p_{m,k}$ with a neural network f_ϕ and obtaining the mixture weights $\pi_{m,k}$ based on the posterior inference results, we apply the neural network f_ϕ to compute spatially dependent mixture weights $\pi_{m,k}$ and an additional network g_ψ to model spatially independent parameters of $p_{m,k}$. In a sense, the neural network f_ϕ acts like learnable priors of spatial dependencies in our method. Mixture weights are regularized by both the intrinsic structure and learnable parameters of the neural network. In the following, we first introduce our approach in a general way and then make a concrete example for clarity.

Truncated Stick-Breaking Mixture Weights

The spatially dependent mixture weights $\pi_{m,k}$ are computed based on outputs of a neural network f_ϕ and are constrained

by the structure and parameters of the network. Let K be the number of components, \mathbf{s}_k the hidden state vector of the k th component, and $\sigma(\cdot)$ the sigmoid function. $c_{k,m}$ is defined as $\sigma(f_\phi(\mathbf{s}_k)_m)$. The computation of $\pi_{m,k}$ which shares a similar form to the truncated stick-breaking process gives

$$\pi_{m,k} = \begin{cases} c_{k,m} \prod_{k' < k} (1 - c_{k',m}), & k < K \\ \prod_{k' < K} (1 - c_{k',m}), & k = K \end{cases} \quad (5)$$

(5) differs from the truncated stick-breaking process in that the length of stick $c_{k,m}$ is computed by a neural network instead of sampled from the beta distribution. Only the first $K - 1$ components are assigned with hidden state vectors which describe the shapes of foreground objects. Complex-shaped background regions are estimated based on the predicted shapes of these foreground objects.

Applying the softmax function to the outputs of neural networks is another viable way of modeling probabilities. The main consideration of choosing (5) over the softmax function is *compositionality*. The purpose of applying the mixture model is to decompose the complex visual scene into simpler components. Generally speaking, objects in the visual scene are relatively easy to represent. The background pixels, however, are almost as hard to model as the visual scene due to the diverse combinations of foreground objects. The softmax function treats each component equally, and $K - 1$ simple and 1 complex hidden state vectors are required to generate the mixture weights for K components. When using (5), only $K - 1$ hidden state vectors are needed. The complex mixture weights of the background component is determined by these simple states vectors.

Conditional Probabilities

The conditional probabilities $p_{m,k}$ are assumed to be spatially independent in the proposed method. Let θ_k^c be the parameters of conditional probabilities $p_{m,k}$. If no constraint is added to θ_k^c , the optimal solution that maximizes the Q-function can be obtained by setting the derivative of (2) with respect to θ_k^c to zero. To lower the complexity of backpropagation and stabilize the training of neural networks, values of θ_k^c are not updated using the closed-form solution. All θ_k^c with $k < K$ are outputs of a neural network g_ψ with hidden states \mathbf{s}_k as inputs. Parameters θ_K^c for the background component are modeled as learnable variables that are optimized by gradient descent in the M-step of the EM algorithm.

Optimization of Mixture Model Parameters

Hidden state vectors \mathbf{s}_k can be updated by either gradient descent with a learnable learning rate or an RNN which imitate the behavior of gradient descent. For the sake of notational simplicity, outputs of the neural networks $f_\phi(\mathbf{s}_k)_m$ and $g_\psi(\mathbf{s}_k)$ are denoted by $f_{k,m}$ and g_k , respectively. If using gradient descent as the update rule, hidden states \mathbf{s}_k with $1 \leq k < K$ are updated by

$$\mathbf{s}_k^{(t+1)} = \mathbf{s}_k^{(t)} + \eta_s \sum_m \left(\gamma_{m,k}^{(t)} \frac{\partial \log p_{m,k}}{\partial g_k} \frac{\partial g_k}{\partial \mathbf{s}_k} + \sum_{k' \geq k} \gamma_{m,k'}^{(t)} \frac{\partial \log \pi_{m,k'}}{\partial f_{k,m}} \frac{\partial f_{k,m}}{\partial \mathbf{s}_k} \right) \Big|_{\mathbf{s}_k = \mathbf{s}_k^{(t)}} \quad (6)$$

If applying an RNN to learn the procedure of gradient descent, inputs to the RNN are features extracted from the concatenation of $\sum_m \gamma_{m,k}^{(t)} (\partial \log p_{m,k} / \partial g_k)$ and all $\sum_{k' \geq k} \gamma_{m,k'}^{(t)} (\partial \log \pi_{m,k'} / \partial f_{k,m})$ with $1 \leq m \leq M$ by an encoder network.

The update rule for the parameters θ_K^c is given by

$$\theta_K^c{}^{(t+1)} = \theta_K^c{}^{(t)} + \eta_\theta \sum_m \gamma_{m,K}^{(t)} \frac{\partial \log p_{m,K}}{\partial \theta_K^c} \Big|_{\theta_K^c = \theta_K^c{}^{(t)}} \quad (7)$$

Learning of Neural Networks

Neural networks f_ϕ and g_ψ are trained by BPTT algorithm. The loss function is chosen as

$$L = - \sum_m \sum_k \gamma_{m,k} (\log p_{m,k} + \log \pi_{m,k}) + \lambda \sum_m D_{\text{KL}}(p_{\text{prior}} || p_{m,K}) \quad (8)$$

As with the loss function (4) used in N-EM, (8) also consists of the Q-function term and a regularization term. This regularization acts like a simplified version of the inductive biases required for human to solve the figure-ground organization problem (distinguishing a figure from the background). In N-EM, prior information of the background p_{prior} is used to regularize the conditional probability distributions of all mixture components. In our method, the conditional distribution of the background component $p_{m,K}$ is encouraged to be similar to p_{prior} , so that the network can better focus on modeling the foreground objects.

Example: Gaussian Conditional Distribution

The conditional probability distribution of a pixel given its component assignment is usually assumed to be Gaussian for real-valued images, and Bernoulli for binary images. We consider only Gaussian distribution in this section (detailed procedures for each image in an epoch are described in Algorithm 1). Formulas for other types of probability distributions can be derived with little modification. Only the means of the Gaussian distributions are learnable parameters, and the variances are assumed to be constant and set to α^{-1} with α being a hyperparameter tuned by cross-validation. Except for the last component, the means of the Gaussian distributions are computed by the neural network g_ψ . For the K th component, it is modeled as a parameter μ_K that is learned by gradient descent. Let x_m be the intensity of the m th pixel. The conditional probability $p_{m,k}$ is computed by

$$p_{m,k} = \begin{cases} \mathcal{G}(x_m; g_\psi(\mathbf{s}_k), \alpha^{-1}), & k < K \\ \mathcal{G}(x_m; \mu_K, \alpha^{-1}), & k = K \end{cases} \quad (9)$$

Substituting (9) and (5) into (6), the gradient descent update rule for hidden states becomes

$$\mathbf{s}_k^{(t+1)} = \mathbf{s}_k^{(t)} + \eta_s \sum_m \left(\alpha \gamma_{m,k}^{(t)} (x_m - g_k^{(t)}) \frac{\partial g_k}{\partial \mathbf{s}_k} + \left(\gamma_{m,k}^{(t)} - \sum_{k' \geq k} \gamma_{m,k'}^{(t)} \sigma(f_{k,m}^{(t)}) \right) \frac{\partial f_{k,m}}{\partial \mathbf{s}_k} \right) \Big|_{\mathbf{s}_k = \mathbf{s}_k^{(t)}} \quad (10)$$

Algorithm 1 Proposed Method (Gaussian Distribution)

```

// Initialization:
Draw  $\mathbf{s}_k^{(0)}$  randomly from  $\mathcal{N}(\mathbf{0}, \sigma_{\text{init}}^2 \mathbf{I})$ ;  $\mu_K^{(0)} \leftarrow \mu_{\text{prior}}$ ;
for  $t \leftarrow 0$  to  $T - 1$  do
  // E-Step:
   $f_{k,m}^{(t)} \leftarrow f_\phi(\mathbf{s}_k^{(t)})_m$ ;  $g_k^{(t)} \leftarrow g_\psi(\mathbf{s}_k^{(t)})$ ;
  Compute  $p_{m,k}^{(t)}$ ,  $\pi_{m,k}^{(t)}$  and  $\gamma_{m,k}^{(t)}$  using (9), (5) and (1);
  // M-Step:
  if using gradient descent then
    Update  $\mathbf{s}_k^{(t+1)}$  using (10);
  else
     $u \leftarrow \alpha \sum_m \gamma_{m,k}^{(t)} (x_m - g_k^{(t)})$ ;
     $v_m \leftarrow \gamma_{m,k}^{(t)} - \sum_{k' \geq k} \gamma_{m,k'}^{(t)} \sigma(f_{k,m}^{(t)})$ ;
     $r_{\text{in}} \leftarrow \text{enc}([u, v_1, v_2, \dots, v_M])$ ;
     $\mathbf{s}_k^{(t+1)} \leftarrow \text{RNN}(r_{\text{in}}, \mathbf{s}_k^{(t)})$ ;
  end if
  Update  $\mu_K^{(t+1)}$  using (11);
  // Loss:
  Compute  $L^{(t)}$  using (8);
end for
// Neural Networks and Learning Rates:
Compute the weighted sum of losses  $L_{\text{sum}} \leftarrow \sum_t w_t L^{(t)}$ ;
Update  $\phi$ ,  $\psi$ ,  $\eta_s$  and  $\eta_\theta$  based on  $L_{\text{sum}}$ ;

```

If updating \mathbf{s}_k with an RNN, the encoder network enc transforms the concatenation of $\alpha \sum_m \gamma_{m,k}^{(t)} (x_m - g_k^{(t)})$ and $\gamma_{m,k}^{(t)} - \sum_{k' \geq k} \gamma_{m,k'}^{(t)} \sigma(f_{k,m}^{(t)})$, $1 \leq m \leq M$ to features r_{in} as the inputs to the RNN.

Substituting (9) into (7) and replacing θ_K^c with μ_K , the learnable parameter for the last component is updated by

$$\mu_K^{(t+1)} = \mu_K^{(t)} + \eta_\theta \alpha \sum_m \gamma_{m,K}^{(t)} (x_m - \mu_K^{(t)}) \quad (11)$$

Related Work

Several approaches have been proposed to solve the perceptual grouping problem and related tasks in recent years. Tagger (Greff et al. 2016) combines the iterative amortized grouping (TAG) mechanism and the Ladder Network (Rasmus et al. 2015) to learn perceptual grouping in an unsupervised manner. It utilizes multiple copies of the same neural network to model different groups in the visual scene and iteratively refine the reconstruction result. RTagger (Prémont-Schwarz et al. 2017) replaces the Ladder Network with the Recurrent Ladder Network and extends Tagger to sequential data. Neural Expectation Maximization (N-EM) (Greff, van Steenkiste, and Schmidhuber 2017) tackles the problem based on the Expectation-Maximization (EM) framework (Dempster, Laird, and Rubin 1977), and achieves comparable performance to Tagger with much fewer parameters. Relational Neural Expectation Maximization (R-NEM) (van Steenkiste et al. 2018) integrates N-EM with a type of Message Passing Neural Network (Gilmer et al. 2017) to learn common-sense physical reasoning based on the compositional object-representations extracted by N-EM.

Our method is mostly related to N-EM. Same as N-EM, the update rules for the distributed representations of objects in the visual scene are derived from the EM algorithm. Different from N-EM, the foreground objects and background are modeled separately in our method, and spatial dependencies of pixels are modeled completely by mixture weights.

Experiments

Datasets Our method is evaluated on two datasets derived from a set of publicly released perceptual grouping datasets provided by (Greff, Srivastava, and Schmidhuber 2015; Greff et al. 2016; Greff, van Steenkiste, and Schmidhuber 2017). We refer to these two datasets as the *Multi-Shapes* dataset and the *Multi-MNIST* dataset. The Multi-Shapes dataset consists of 4 subsets, which differs from each other in either the image size (20×20 or 28×28) or the number of objects in each image (2, 3 or 4). Each subset contains 70,000 binary images with simple shapes located at random positions. The Multi-MNIST dataset is composed of 3 subsets with different degrees of object variations. In each subset, there are 70,000 grayscale 48×48 images which contain 2 handwritten digits. Images in both Multi-Shapes and Multi-MNIST datasets may contain overlapped objects.

Comparison Methods The proposed method is compared with two state-of-the-art perceptual grouping algorithms, namely, *N-EM* (Greff, van Steenkiste, and Schmidhuber 2017) and *Tagger* (Greff et al. 2016). To assess the effectiveness of computing mixture weights using (5), we substitute it with the softmax function and evaluate this modification (referred as *softmax method*) on the Multi-Shapes dataset. Neural networks are trained with the Adam optimization algorithm (Kingma and Ba 2015) for all approaches. To make a fair comparison, we do not specially design the neural networks f_ϕ and g_ψ for the proposed method. The structure of f_ϕ which generates mixture weights in our method is identical to the network f_ϕ in N-EM. The extra g_ψ for computing conditional probabilities is simply a one-layer fully-connected neural network, in which the number of parameters is equal to the dimension of hidden states and negligible compared to the total number of network parameters.

Evaluation Metrics We use two metrics to evaluate the grouping results. The first is *Adjusted Mutual Information* (AMI), which measures the accuracy of mixture assignments. To be consistent with previous work (Greff, Srivastava, and Schmidhuber 2015; Greff et al. 2016; Greff, van Steenkiste, and Schmidhuber 2017), AMI scores are not computed in the background and overlapping regions. The second is *Mean Squared Error* (MSE), which computes the smallest mean squared differences between the estimated images of individual objects and the ground truth images over all possible permutations (formulated as the assignment problem and solved efficiently by the Hungarian method). MSE scores are evaluated at all pixels.

Perceptual Grouping Performance Comparisons

Performances of Tagger, N-EM and the proposed method are compared on two subsets of the Multi-Shapes dataset which

differ from each other in image size (20×20 and 28×28). Images in both subsets contain 3 objects. Because objects in both subsets are of the same size, 20×20 images which consist of less pixels are more likely to contain overlapped objects, and performing perceptual grouping on them is not necessarily easier. For Tagger, parameters of mixture models are updated via a 3-layer Ladder Network. For N-EM and the proposed method, the encoder and decoder networks are convolutional neural networks (CNNs) with 2 convolutional, 2 fully-connected and 3 layer normalization (Ba, Kiros, and Hinton 2016) layers. The qualitative results are presented in Figure 1, and the AMI/MSE scores are presented as follows.

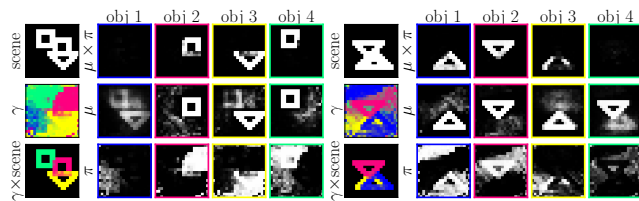
	Size 20×20	Size 28×28
Tagger	0.933 / 0.71e-2	0.820 / 1.49e-2
N-EM	0.824 / 1.59e-2	0.897 / 0.82e-2
Proposed	0.920 / 0.40e-2	0.941 / 0.28e-2

On the subset consisting of 20×20 images, Tagger achieves the highest grouping accuracy (AMI) and the proposed method best estimates the images of individual objects (MSE). The AMI score of the proposed method is slightly lower than Tagger. On the other subset, The proposed method achieves both the best AMI and MSE scores.

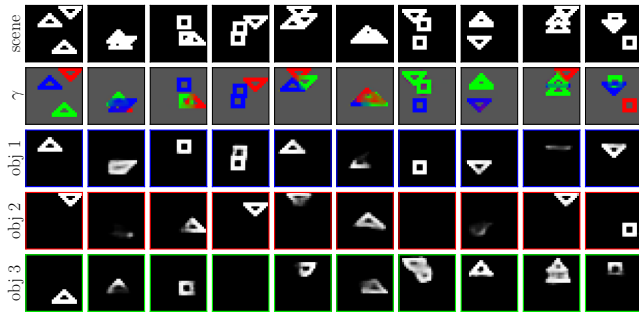
According to Figure 1a (*Tagger*), the learned knowledge of correlations between pixels are entangled. Both conditional probability distributions (row 2; cols 2–5, 7–10) and mixture weights (row 3; cols 2–5, 7–10) contain partial and imperfect information of shape and appearance. It is caused by modeling both conditional probability distributions and mixture weights to be spatially dependent. Another observation is that objects are mainly distinguished based on conditional distributions, and the mixture weights assist the segregation when ambiguity exists. Although the mixture component assignments are predicted accurately for foreground objects (row3; cols 1, 6), the background pixels are not well inferred based on posterior probabilities γ (row2; cols 1, 6). The possible reason is that the diverse combinations of foreground objects prevent background pixels from being modeled well using non-combinational representations.

In Figure 1b (*N-EM*), the colors of background pixels in the 2nd row are gray (red+green+blue) because the posterior probabilities γ are almost identical for all mixture components. Same as Tagger, background pixels cannot be well determined via inferences of posterior probabilities in N-EM. This is caused by a side effect of solving the entangled learning of spatial dependencies by freezing the mixture weights during the procedures of expectation maximization. Because correlations between pixels are modeled mainly by conditional probability distributions, component assignments are ambiguous in regions where the means of the distributions (rows 3–5) are similar.

As demonstrated in Figure 1c (*proposed method*), objects are well segregated based on the learned knowledge of spatial dependencies even if objects are heavily overlapped (cols 2, 6). Although the reconstructed images of individual objects are not completely correct, by regularizing the mixture weights with the deep priors learned from data, the predicted results follow the principles of proximity, similarity, continuity, and closure in the Gestalt laws of grouping.



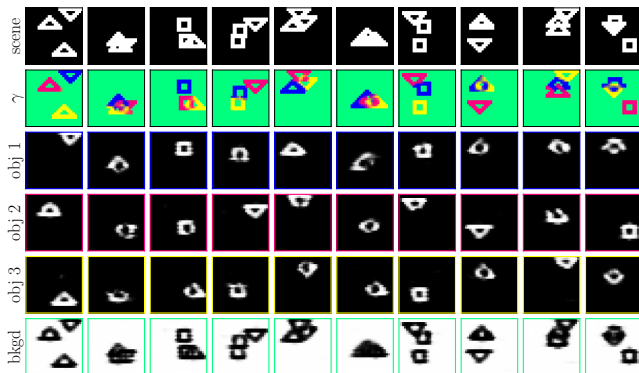
(a) Tagger (Size 20×20)



(b) N-EM (Size 28×28)



(c) Proposed (Size 28×28)



(d) Softmax (Size 28×28)

Figure 1: Component assignments and reconstructed images of individual objects evaluated on the Multi-Shapes dataset.

Effectiveness of Modeling Background Separately

Besides the observation from Figure 1a that modeling the background as an ordinary component results in imperfect separation of foreground and background based on posterior probabilities, we also assess the effectiveness of modeling

	F	N	N-EM		Proposed	
			AMI	MSE	AMI	MSE
Bernoulli	16	16	0.876	$0.95e-2$	0.856	$1.28e-2$
	32	16	0.897	$0.82e-2$	0.868	$1.23e-2$
	32	32	0.816	$1.32e-2$	0.926	$0.43e-2$
	64	32	0.814	$1.34e-2$	0.920	$0.40e-2$
	64	64	0.749	$1.70e-2$	0.941	0.28e-2
Gaussian	16	16	0.687	$2.35e-2$	0.915	$0.68e-2$
	32	16	0.676	$2.34e-2$	0.911	$0.63e-2$
	32	32	0.566	$2.87e-2$	0.914	$0.51e-2$
	64	32	0.503	$3.06e-2$	0.935	$0.30e-2$
	64	64	0.353	$3.77e-2$	0.939	0.29e-2

Table 1: Comparisons of AMI and MSE scores under different choices of conditional probability distributions. F and N are respective dimensions of RNN inputs and hidden states.

the background separately by evaluating the performances of the softmax method on the second subset (size 28×28) used in the previous experiment. The best AMI and MSE scores achieved are 0.875 and 1.10×10^{-2} , which are significantly worse than the results of the proposed method (0.941 and 0.28×10^{-2}). The burden of representing the complex background with a single hidden state vector prevents the model to focus on spatial dependencies of foreground pixels. The qualitative results of the softmax method is shown in Figure 1d. The estimated images of individual objects are not visually plausible although the results of posterior inferences (row 2) are satisfactory.

Generalizability and Sensitivity Analyses

Generalizability of Compositional Representations To illustrate the generalizability of the learned compositional representations, the proposed method are evaluated on three subsets of the Multi-Shapes dataset that differ in the number of objects in images. Results of N-EM are also presented. Models of both methods are trained *only* on the second subset. The AMI/MSE scores are shown below.

	N-EM	Proposed
2 Objects	$0.906 / 0.98e-2$	$0.961 / 0.11e-2$
3 Objects (Train)	$0.897 / 0.82e-2$	$0.941 / 0.28e-2$
4 Objects	$0.838 / 1.44e-2$	$0.901 / 0.69e-2$

For both approaches, the learned representations of conceptual entities generalize well to visual scenes containing more or less objects. The accuracies of estimated mixture component assignment are slightly higher when the task is simpler, and moderately lower when dealing with more challenging visual scenes. The MSE score of N-EM becomes slightly worse on the simpler task.

Sensitivity to Choice of Conditional Distributions We assess the influences of different choices of conditional probability distributions on the second subset (3 objects) used in the previous experiment. Detailed performances of N-EM and the proposed method are shown in Table 1. If intensities of binary images are modeled by Gaussian distributions, the grouping results of N-EM deteriorate drastically.

	F	N	N-EM		Proposed	
			AMI	MSE	AMI	MSE
#1	32	16	0.780	1.17e-2	0.748	1.89e-2
	32	32	0.573	1.54e-2	0.765	1.64e-2
	64	32	0.521	1.54e-2	0.787	1.35e-2
	64	64	0.391	1.79e-2	0.790	1.30e-2
#2	32	16	0.317	2.86e-2	0.719	2.07e-2
	32	32	0.254	2.75e-2	0.704	2.04e-2
	64	32	0.194	2.71e-2	0.719	1.61e-2
	64	64	0.152	2.73e-2	0.676	1.63e-2
#3	32	16	0.335	2.93e-2	0.715	2.16e-2
	32	32	0.223	3.15e-2	0.702	2.07e-2
	64	32	0.185	2.94e-2	0.705	1.69e-2
	64	64	0.158	2.84e-2	0.683	1.62e-2

Table 2: AMI and MSE scores evaluated on different subsets of the Multi-MNIST dataset. F and N are respective dimensions of RNN inputs and hidden states.

The proposed method, on the other hand, are less sensitive to the form of conditional distributions.

Sensitivity to Diversity of Objects The Multi-MNIST dataset is much more challenging than the Multi-Shapes dataset for the larger image size (48×48 versus 20×20 or 28×28), as well as higher degrees of both inter-object variations (the number of unique objects) and intra-object variations (intensities of pixels belonging to the same object). There are 3 subsets in this dataset. Subsets 1, 2 and 3 are constructed using 20, 500 and 70,000 unique digits, respectively. Different from the first two subsets, the digits composing images in the test set of subset 3 do not exist in the training and validation set. We evaluate the performances of N-EM and the proposed method on these subsets, and compare the sensitivities to object diversities of the two approaches. Experimental results are presented in Table 2.

When the diversity of objects is small, N-EM and the proposed method perform similarly well. N-EM achieves slightly higher AMI score and the proposed method attains lower MSE score. As the variations of objects increase, the performance of the proposed method drops moderately, and the grouping results of N-EM decline significantly. On subsets 2 and 3, both methods achieve the highest AMI scores when dimensions of RNN inputs and hidden states are small, and the best MSE scores when these dimensions are slightly larger. The possible reason is that the higher dimensions allow models to focus on more details and help reconstruct sharper individual digits. The large diversities of objects, on the other hand, misguide models to represent parts of different objects in the same mixture component and lower the grouping accuracies as the capacities of models increase.

Figure 2 demonstrates the predicted component assignments and reconstructed images of individual objects produced by both methods. On subset 1, the results generated by N-EM are slightly more visually plausible. Except for the occluded regions, details of each object are accurately reconstructed. This is because N-EM allows the conditional

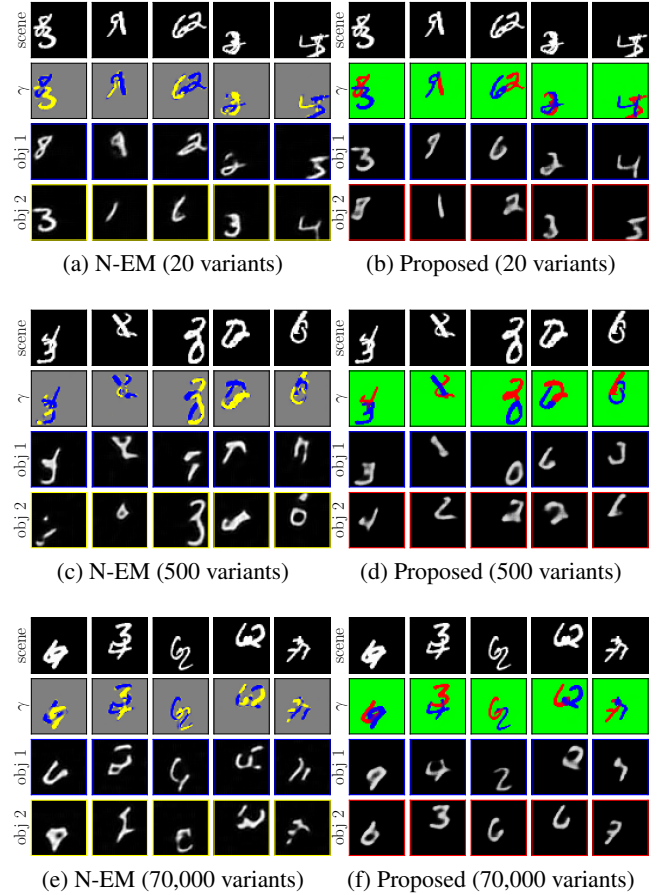


Figure 2: Component assignments and reconstructed images of individual objects evaluated on the Multi-MNIST dataset.

probabilities distributions to vary within each object, which helps better reconstruct the intra-object variations of each digit. Both methods can accurately segregate objects from some complex visual scenes like the last two images in Figure 2a and 2b. On subsets 2 and 3, the estimated images of individual objects generated by N-EM are also sharper than the proposed method. However, each component contains pixels of multiple objects. The proposed method models the spatial dependencies of pixels with prior knowledge of mixture weights, and utilizes the principle of similarity implied by the spatially independent conditional distributions to assist the learning of correlations between pixels. Empirical results suggest that this scheme is less sensitive to the diversities of objects in the visual scenes.

Conclusion

In this paper, we have proposed a type of novel spatial mixture models which place learnable deep priors on the mixture weights for tackling the perceptual grouping problem. By modeling the background as a special component in a compositional manner and decomposing the descriptions of conceptual entities into shapes and appearances that are

separately represented, the proposed method better learns spatial dependencies than existing methods for perceptual grouping. We have demonstrated that the proposed method is insensitive to diversities of objects, and the learned compositional representations of individual entities generalize well to visual scenes constructed by novel combinations of these entities. Further research in this direction could be integrating hierarchical spatial mixture models with neural networks and solving higher-level tasks like relational inferences based on the learned compositional representations.

Acknowledgments

This work was supported in part by National Key R&D Program of China (No.2017YFC0803700), NSFC under Grant No.61572138 & No.U1611461, and STCSM Projects under Grant No.16JC1420400 & No.18511103104.

References

- Ba, J. L.; Kiros, J. R.; and Hinton, G. E. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.
- Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(8):1798–1828.
- Biederman, I. 1987. Recognition-by-components: A theory of human image understanding. *Psychological Review* 94(2):115.
- Bienenstock, E.; Geman, S.; and Potter, D. 1997. Compositionality, MDL priors, and object recognition. In *Advances in Neural Information Processing Systems (NIPS)*, 838–844.
- Dempster, A. P.; Laird, N. M.; and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 1–38.
- Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; and Dahl, G. E. 2017. Neural message passing for quantum chemistry. In *International Conference on Machine Learning (ICML)*, 1263–1272.
- Goldstein, E. B., and Brockmole, J. 2016. *Sensation and perception*. Cengage Learning.
- Greff, K.; Rasmus, A.; Berglund, M.; Hao, T.; Valpola, H.; and Schmidhuber, J. 2016. Tagger: Deep unsupervised perceptual grouping. In *Advances in Neural Information Processing Systems (NIPS)*, 4484–4492.
- Greff, K.; Srivastava, R. K.; and Schmidhuber, J. 2015. Binding via reconstruction clustering. *arXiv preprint arXiv:1511.06418*.
- Greff, K.; van Steenkiste, S.; and Schmidhuber, J. 2017. Neural expectation maximization. In *Advances in Neural Information Processing Systems (NIPS)*, 6691–6701.
- Grossberg, S.; Mingolla, E.; and Ross, W. D. 1997. Visual brain and visual perception: How does the cortex do perceptual grouping? *Trends in Neurosciences* 20(3):106–111.
- Hummel, J. E., and Biederman, I. 1992. Dynamic binding in a neural network for shape recognition. *Psychological Review* 99(3):480.
- Kingma, D. P., and Ba, J. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.
- Lake, B. M.; Ullman, T. D.; Tenenbaum, J. B.; and Gershman, S. J. 2017. Building machines that learn and think like people. *Behavioral and Brain Sciences* 40.
- Milner, P. M. 1974. A model for visual shape recognition. *Psychological review* 81(6):521.
- Prémont-Schwarz, I.; Ilin, A.; Hao, T.; Rasmus, A.; Boney, R.; and Valpola, H. 2017. Recurrent ladder networks. In *Advances in Neural Information Processing Systems (NIPS)*, 6009–6019.
- Rao, A. R.; Cecchi, G. A.; Peck, C. C.; and Kozloski, J. R. 2008. Unsupervised segmentation with dynamical units. *IEEE Transactions on Neural Networks* 19(1):168–182.
- Rasmus, A.; Berglund, M.; Honkala, M.; Valpola, H.; and Raiko, T. 2015. Semi-supervised learning with ladder networks. In *Advances in Neural Information Processing Systems (NIPS)*, 3546–3554.
- Reichert, D. P., and Serre, T. 2014. Neuronal synchrony in complex-valued deep networks. In *International Conference on Learning Representations (ICLR)*.
- Robinson, A., and Fallside, F. 1987. *The utility driven dynamic error propagation network*. University of Cambridge Department of Engineering.
- Treisman, A. 1996. The binding problem. *Current Opinion in Neurobiology* 6(2):171–178.
- van den Hengel, A.; Russell, C.; Dick, A.; Bastian, J.; Pooley, D.; Fleming, L.; and Agapito, L. 2015. Part-based modelling of compound scenes from images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 878–886.
- van Steenkiste, S.; Chang, M.; Greff, K.; and Schmidhuber, J. 2018. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. In *International Conference on Learning Representations (ICLR)*.
- Wang, D., and Terman, D. 1995. Locally excitatory globally inhibitory oscillator networks. *IEEE Transactions on Neural Networks* 6(1):283–286.
- Werbos, P. J. 1988. Generalization of backpropagation with application to a recurrent gas market model. *Neural Networks* 1(4):339–356.
- Wolfe, J. M., and Cave, K. R. 1999. The psychophysical evidence for a binding problem in human vision. *Neuron* 24(1):11–17.