Learning GMRF Structures for Spatial Priors

Lie Gu, Eric P. Xing and Takeo Kanade
Computer Science Department
Carnegie Mellon University
{gu, epxing, tk}@cs.cmu.edu

Abstract

The goal of this paper is to find sparse and representative spatial priors that can be applied to part-based object localization. Assuming a GMRF prior over part configurations, we construct the graph structure of the prior by regressing the position of each part on all other parts, and selecting the neighboring edges using a Lasso-based method. This approach produces a prior structure which is not only sparse, but also faithful to the spatial dependencies that are observed in training data. We evaluate the representation power of the learned prior structure in two ways: first is drawing samples from the prior, and comparing them with the samples produced by the GMRF priors of other structures; second is comparing the results when applying different priors to a facial components localization task. We show that the learned graph captures meaningful geometrical variations with significantly sparser structure and leads to better parts localization results.

1. Introduction

Visual objects from the real world can often be represented in terms of a set of parts or landmarks arranged in a deformable configuration. In a typical Bayesian formulation of many visual recognition problems, the *a priori* geometric relationships among object parts in all possible deformations can often be expressed by a prior distribution over the locations of the these parts. Such a distribution is usually known as a *spatial prior*.

The Gaussian Markov Random Fields (GMRF) is widely used as a natural model in computer vision for modelling spatial priors. Under a GMRF, a priori the (coordinates of) parts in an (deformable) object follows a multivariate Gaussian distribution. The graph structure underlying a GMRF encodes the assumed spatial relationships among parts, and hence it also determines the computational complexity while performing detection or recognition tasks. The dependency structures of the spatial priors used in previous vision recognition approaches vary dramatically from fully connected joint Gaussian graphs [5] to decomposable star-structured graph [1], tree-structured graph [4], and fully disconnected graph [11]. A major motivation for choosing models with sparse dependencies is the resulting saving in computational cost, but constructing such "sparse models" requires substantial manual engineering. For example, the recently proposed star-structured model known as *k-fan* [1] is constructed by first selecting a number of key points as the reference parts, and then decomposing the joint spatial prior of all parts based on the assumption that given the reference parts, all other object parts are conditional independent. Although these types of simplifying assumptions can lead to a highly efficient recognition algorithm, the resulting graph structures could severely limit the representation power of the prior models because of the possibly artificial constraints on part-deformation they enforce. In this paper, we depart from the aforementioned heuristic way of *designing* a sparse spatial prior, as practiced in almost extant recognition algorithms known to us, and adopt a methodology of automatically *learning* a sparse spatial prior from landmarked objects, that can faithfully capture the key geometrical regularities and spatial constrains of object parts revealed in the training data.

The sparsity of the learned spatial priors is enforced by using a Lasso based approach [12] which was originally proposed by Tibshirani as a variable selection method. Lasso is extensively used in linear regression because it produces interpretable models. Dobra *et.al,* [2] and Meinshausen *et.al,* [10] first applied Lasso regression to the structure learning problem in GMRFs. Inspired by their success in constructing sparse, large-scale graphs, we apply this approach to learn the structure of spatial prior for visual objects. The resultant structures turn to be very sparse, as we will show in following sections. We evaluate the representation power of the learned graph structures by drawing samples from the corresponding priors, and comparing them with the samples drawn from other priors with pre-specified structures. From these samples we observe that the learned prior structures preserve as much meaningful spatial variations as the fully connected graphs. Al-

though the learned graphs are not necessarily decomposable, we can still take the advantage of the graph sparseness to speed up object localization. We evaluate the capacity of the learned graph structures using a greedy search algorithm that incorporates the spatial priors with image evidences to locate an object and its parts in test images. By iterating and maximizing the conditional density of randomly chosen local graph structures, the algorithm maximizes the posterior of the object spatial configuration in a greedy manner. The computation cost of this algorithm is decided by the sparseness of the prior structures.

In the next section we will introduce the basic setting of the problem. We describe the Lasso-based approach for structure learning and the algorithm for object localization in Sections 3.1 and 3.2. Experimental results are shown and explained in Section 4.

2. Problem Setting

The geometrical structure of an object of interest is usually described by its part positions in a vector form,

$$V = (v_1, \dots, v_d) \tag{1}$$

where $v_i = (x_i, y_i)$ denotes the image plane coordinate of the ith part. We seek to construct a graphical model representation, in which the parts are modelled by graph vertices and the spatial relations of the parts are modelled by graph edges. Given a collection of manually labelled or automatically obtained training shapes $\{V^1, \dots V^N\}$, our interest is to find the best graph structure among all models. This is a non-trivial model selection problem. The desired graph needs to be representative and sparse, so that it captures all significant geometrical deformations presented in training data, yet restricts the computation cost at a relatively low level while performing object recognition tasks.

Before we proceed forward, we need to clarify that there are two types of geometry information contained in the shape vector V: the *canonical* object shape which is invariant to geometrical transforms, and the object pose which instantiates the canonical object in an image. For the purpose of modelling part relations we will only be interested in the former type of information. We use generalized procrustes analysis [6] to compute a reference shape V_o from training shapes, and normalize it by zero centroid and unit norm. We align each training shape V^i to V_o by solving a rigid transform $\hat{\Gamma}^i$ that minimizes the difference error,

$$\hat{\Gamma}^i = \mathrm{argmin}_\Gamma \|V_o - \Gamma(V^i)\| \tag{2}$$

so the centroid, size and orientation of the shape $\hat{\Gamma}^i(V^i)$ are normalized accordingly. We call $\hat{\Gamma}(V)$ the *canonical shape* and refer it as V in the following sections.

We model the canonical shape V by a Gaussian Markov Random Field model $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, which allows the shape to deform according to a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma)$. Let $Q = \Sigma^{-1}$ denote the precision matrix or the concentration matrix. The graph structure is fully determined by the nonzero pattern of Q. That is, $Q_{ij} \neq 0$ if and only if $e_{ij} \in \mathcal{E}$. Equivalently, we have

$$\rho_{ij} \neq 0 \Leftrightarrow e_{ij} \in \mathcal{E} \tag{3}$$

where $\rho_{ij} \triangleq \mathrm{Corr}(V_i, V_j | V_{-ij}) = -Q_{ij}/\sqrt{Q_{ii}Q_{jj}}$ denotes the ij-th partial correlation, i.e., the conditional correlation of the part V_i and the part V_j given all other parts V_{-ij}.

3. Learning Spatial Structure for Object Recognition

3.1. Structures Learning by Lasso Regression

The standard approach to structure learning in GMRFs is pruning edges from a fully connected graph by testing the elements of the sample precision matrix Σ^{-1}, such as the stepwise method [7] or the simultaneous testing method [3]. However, these methods are not well suited for large graphs or graphs with singular empirical covariance structures. In object recognition, placing landmarks evenly along the contours of objects usually causes strong linear dependencies among their positions, i.e., $AV = e$, where $e \sim \mathcal{N}(\mu_e, \Sigma_e), \Sigma_e \to 0$. And the number of training samples could be less than the dimension of landmark vectors. To avoid directly computing Σ^{-1} in these cases we adopt a lasso regression based method proposed by Meinshausen et.al. [10] and Dobra et.al. [2].

The regression of one node V_i on all other nodes V_{-i} in the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is given by,

$$\hat{\theta}_i = \underset{\theta_i}{\mathrm{argmin}}\, E\left(V_i - \sum_{k \in \{1..d\} \setminus i} V_k \theta_i^k\right)^2 \tag{4}$$

Note that the elements of $\hat{\theta}_i$ are determined by the precision matrix Q [7] such as

$$\hat{\theta}_i^k = -Q_{ki}/Q_{kk} \tag{5}$$

Therefore $e_{ij} = 0$ is equivalent to $\hat{\theta}_i^j = 0$. In other words, the set of non-zero coefficients of θ_i determines the neighbors of the i-th node in the graph. This motivates the use of the lasso regression method to construct a sparse graph by minimizing the l_1-penalized error,

$$\hat{\theta}_i^\lambda = \underset{\theta_i}{\mathrm{argmin}}\, E\left(V_i - V_{-i}\theta_i^{-i}\right)^2 + \lambda_i \sum_k |\theta_i^k| \tag{6}$$

The edge estimate between node V_i and V_j is defined by $e_{ij} \in \mathcal{E} : \hat{\theta}_{i,j}^\lambda \neq 0 \;\; \hat{\theta}_{j,i}^\lambda \neq 0$. Hence each choice of a

penalty parameter λ specifies an estimate of graph structure. Meinshausen and Buhlmann [10] show that cross-validation dose not lead to a good choice of λ. In practice we can use

$$\lambda_i^* = \frac{2\sigma_i}{\sqrt{n}}\Phi^{-1}\left(1-\alpha/2d^2\right) \quad (7)$$

where $\alpha \in [0,1]$ controls overall confidence level.

3.2. Object Recognition Using a Sparse GMRF

Now we consider the problem of localizing objects and their parts in a new testing image. Suppose that the posterior probability of observing the object at a particular spatial configuration $V = \{v_1, \ldots, v_d\}$ on the image I is given by $p(V|I)$. The image evidence $p_i(I|v_i)$ of seeing the i-th part at the position v_i is assumed to be independent to other parts given their spatial locations. According to this independence assumption, the posterior $p(V|I)$ can be factorized as,

$$p(V|I) \propto p(V,I) = p(v_1,\ldots,v_d)\prod_{i=1}^{d} p_i(I|v_i) \quad (8)$$

Here $p(v_1,\ldots,v_d)$ is the spatial prior and its graph structure of the corresponding GMRF model is learned by lasso regression.

The image evidence $p_i(I|v)$ of a part is modelled by its local appearance. For the ith part, we sample a small image patch centered on v_i, compute the magnitude and the orientation of the intensity gradient for every pixel within the patch, and stack them into a histogram vector F_i as described in [8]. We model $p_i(F_i|v_i)$ using a Mixture of Gaussian (MoG), and learn the model parameters from a set of labelled training images.

The feature detector is run on the image independently for every part. Suppose for the ith part we find m candidate positions $\{v_i^1, \ldots, v_i^m\}$ by computing the modes of the density $p_i(F_i|v_i)$. We formulate the problem of object recognition as finding the best configuration for all d parts. A trivial resort to maximizing the posterior $p(V|I)$ over all possible configurations $\{v_i^j : i = 1..d; j = 1..m\}$ is usually computationally infeasible ($\mathcal{O}(m^d)$) as d grows large. Instead, we propose an alternative algorithm that maximizes the posterior $p(V|I)$ in a greedy way.

1. Initialize $\{v_1, \ldots, v_d\}$ randomly using their candidate positions.
2. Pick one part v_i, $i \in [1..d]$ randomly and select its neighboring parts $\mathcal{N}(v_i)$.
3. Fix the positions of $\mathcal{N}(v_i)$, find the best position of v_i from its candidates by maximizing the conditional probability,
$$p(F_i|v_i)p(v_i|\mathcal{N}(v_i)) \quad (9)$$
4. Repeat steps 2 and 3 until convergence.

In each iteration, the algorithm maximizes the conditional density $p(v_i|I, \mathcal{N}(v_i)))$ within a randomly picked local graph structure $\{v_i, \mathcal{N}(v_i)\}$. This greedy maximization strategy can effectively reduce the computation complexity while the spatial prior is encoded within a sparse graph. Since the conditional variance of $\mathrm{Var}(v_i|\mathcal{N}(v_i))$ can be computed beforehand in the training phrase, the computational cost is determined by calculating the conditional mean,

$$\mu_{v_i|\mathcal{N}(v_i)} = \mu_{v_i} + \Sigma_{v_i,\mathcal{N}(v_i)}\Sigma^{-1}_{\mathcal{N}(v_i)}(\mathcal{N}(v_i) - \mu_{\mathcal{N}(v_i)}) \quad (10)$$

So the overall computation complexity of the object localization algorithm is reduced from $\mathcal{O}(m^d)$ to $\mathcal{O}(kmd)$, while k is the number of iterations. The convergence of the algorithm is not guaranteed while the graph structure contains circles or loops. In practice we resort to multiple initialization, and compare the global posterior density to ensure its increase.

The algorithm selects the best spatial configuration $\hat{V} = \{\hat{v}_1, \ldots, \hat{v}_d\}$ from a discrete set of candidate positions $\{v_i^j : i = 1..d; j = 1..m\}$ given by the density modes of $p(I|v_i)$. Next we adopt a de-nosing step to deal with miss matchings in $\{\hat{v}_1, \ldots, \hat{v}_d\}$: first project the vector $\hat{V}$ into its principle subspace; preserve the first several eigen-components, and reconstruct the vector using these components only. This step will enforce to correct the miss matching parts in $\hat{V}$ and move them to "reasonable good" positions. Finally we refine $\hat{V}$ by adjusting each $\hat{v}_i$ continuously within a small local region using its image evidence model $p(I|v_i)$.

4. Experimental Results

We apply the presented approach to learn the structures of GMRF priors for three types of objects: face, hand and human body. We compare the learned structure with three other different graph structures: the complete graph [5], the k-fan graph [1] and the fully disconnected graph. We first compute the MLE estimates of the model parameters, then draw samples from all graphs and visualize them. Then we compare the accuracy for face and facial parts localization using the algorithm presented in section 3.2.

Our experiments involve three databases: 1) AR face database [9]: 720 frontal face images. Each image contains 83 manually labelled landmarks along the contours of main facial components. 2) USH human body database: 112 indoor video sequences of 28 different walking persons. All videos are taken from side-view and every frame contains 74 landmarks. The landmarks on the first frame of each video sequence are manually labelled, then tracked through the rest frames using a feature point tracker. 3) Hand database: 40 hand images, each image is labelled with 56 landmarks.

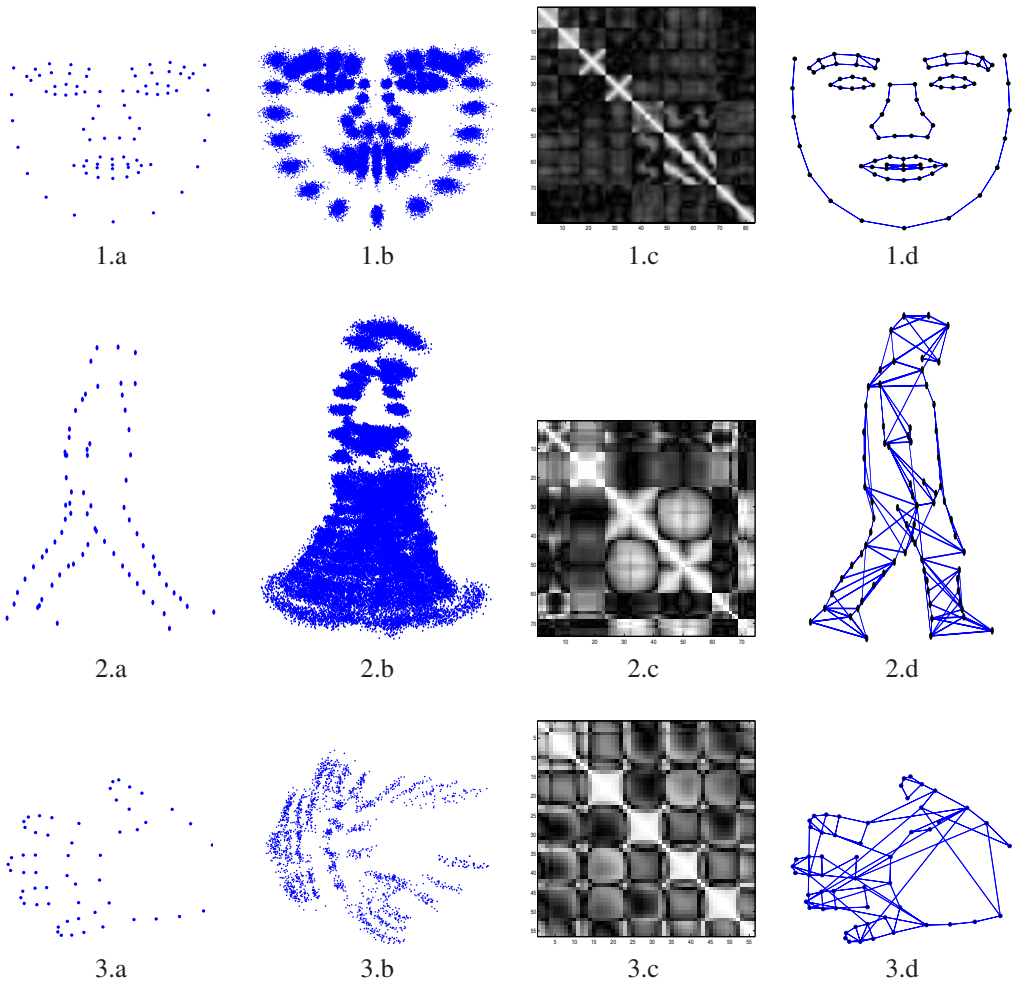

Figure 1. *The prior structures learned for different objects. From top to bottom the figure shows the results on face, human body and hand; from left to right the subfigures show: a) the landmarks on the objects; b) the superimposition of training samples; c) the sample correlation matrices; d) the graph structures learned from training samples. Note that there is an interesting coherence between the graph structures determined by statistical dependencies among landmarks and the physical structures of objects.*

4.1. Structure Learning

Figure 1 shows the databases and the learned graph structures. The first column (a) shows the landmarks put on the objects. One way to visualize the variance of the landmarks is by superimposing training samples, as shown in the second column (b). From that we can observe that the marginal distributions of each landmark are approximately normal. The third column (c) shows the sample correlation matrices. Although the correlation matrices appear to be dense, by lasso based variable selection, we can approximate the partial correlation structures (precision matrices) by sparse graphs, as shown in the last column (d). The ordering of landmarks in correlation matrices (c) dose not affect the learning results.

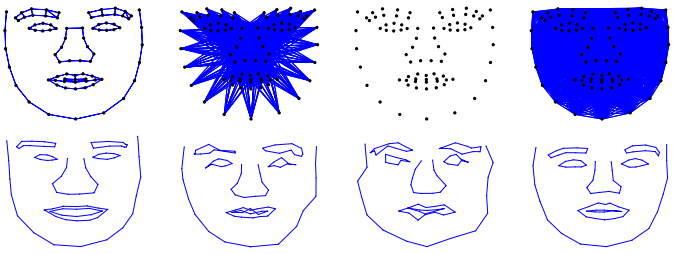

Figure 2. *Four different GMRFs structures (top) and the typical samples (bottom) drawn from each of them. a) the graph learned from data. b) the k-fan graph using nose as the reference part. c) the fully disconnected graph. d) the complete graph. We compute the MLE of model parameters of different structures using the same set of training data.*

4.2. Comparison by Drawing Samples

It is interesting to observe that the statistically learned structures resemble the physical structures of the objects very closely. Although the learning algorithm is not aware of the physical connections among landmarks, many adjacent nodes in the graph are indeed jointed with each other physically. It is natural to ask how effective dose these graphs represent or constrain the geometrical deformations for the corresponding objects.

A straightforward way to compare the representational capacity of different prior models is by looking at the samples drawn from them. Figure 2 shows a typical set of face samples (bottom) drawn from the four types of graphs (top). More samples are shown in Figure 3. We have several observations. Although the structure of the learned graph (a) is considerably sparser than the full graph (d), the samples drawn from (a) are as natural as those drawn from (d). This is because only small partial correlation coefficients are discarded in approximating the precision matrix, so the major spatial dependencies captured in the training landmarks are preserved in the learned graph. In contrast, enforcing a heuristic prior structure on the objects often limits the representation power. The top row of the column (b) shows a k-fan graph. In this particular example, we choose the reference set (the k part) to be the points along the nose contour, therefore we have $k = 12$. That is a very dense graph as shown in the figure. Conditioning on the reference points the k-fan graph assumes all other points are independent to each other. As a result, the other facial components (such as eyes, mouth and silhouette) of the samples drawn from this graph are jaggy and uneven comparing to those in (a). The samples drawn from fully disconnected graph (c) appear to be more irregular due to the lack of constraint among the landmarks.

Figure 3 compares more i.i.d. samples drawn from the four different graph structures. To avoid using fixed landmarks as the reference part in the k-fan graph, we use the method in [1] to select the optimal reference landmarks. The number k is set to be 7 for faces, 6 for human bodies and 4 for hands. This choice leads to relatively dense graphs (for example, $28 + (7 \times 76) = 560$ edges for face compared to 94 edges in Figure 1.d), and a reasonable training time to find the optimal reference set. The drawn samples are shown in Figure 3.b. Based on these observations we conclude that the learned graph structures preserve approximately same amount of representation power as the full graphs with significantly less edges.

4.3. Facial Components Localization

We compare the capacity of the learned graph and k-fan graph in localizing facial components using the approach described in section 3.2 . The experiment is performed on the frontal face database. We use 520 face samples for training and the rest of 200 samples for testing. All testing faces are frontal and upright. The part locations generated by the algorithm were compared with the manually labelled ground truth. Since there exists miss matchings and false alarms in the feature detection results, we computed the 85% trimmed mean of the distance between the localized parts and the ground truth labels for whole face and five individual facial components. Table 1 summarizes the average errors of face and facial component localization results by using two graph structures.

5. Discussion and Future Work

We have described a novel application of structure learning techniques for object recognition, especially, in the sense of using GMRF as a prior for structured data. It is encouraging to see that the learned model is superior in its representation power (fig 2 & 3), sparseness 2, and it leads better recognition results when comparing with the-state-of-art "designed" models.

The algorithm we used for recognition is a coordinate descent algorithm in its nature. We expect that due to the continuous nature of our space for shape priors, Monte Carlo methods may require very long mixing time under our model, which actually conjoins the mixture of Gaussian likelihood model and a GMRF prior. But we are aware of the relative sampling techniques. Our future work will include implementing a Gibbs sampler for the shape model for comparison, and a full Bayes treatment of pose parameters.

References

[1] D. Crandall, P. Felzenszwalb, and D. Huttenlocher. Spatial priors for part-based recognition using statistical models. In *Proceedings of Computer Vision and Pattern Recognition*, 2005.

[2] A. Dobra, C. Hans, B. Jones, J. Nevins, G. Yao, and M. West. Sparse graphical models for exploring gene expression data. *J. Mult. Analysis 90:196.212*, 2004.

[3] M. Drton and M. D. Perlman. A sinful approach to model selection for gaussian concentration graphs. *Biometrika*, 2004.

[4] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *International Jornal of Computer Vision*, 2005.

[5] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of Computer Vision and Pattern Recognition*, 2003.

[6] J. C. Gower. Generalized procrustes analysis. *Psychometrika*, 1975.

[7] S. Lauritzen. *Graphical models*. Oxford University press, New York, 1996.

[8] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Jornal of Computer Vision*, 2004.

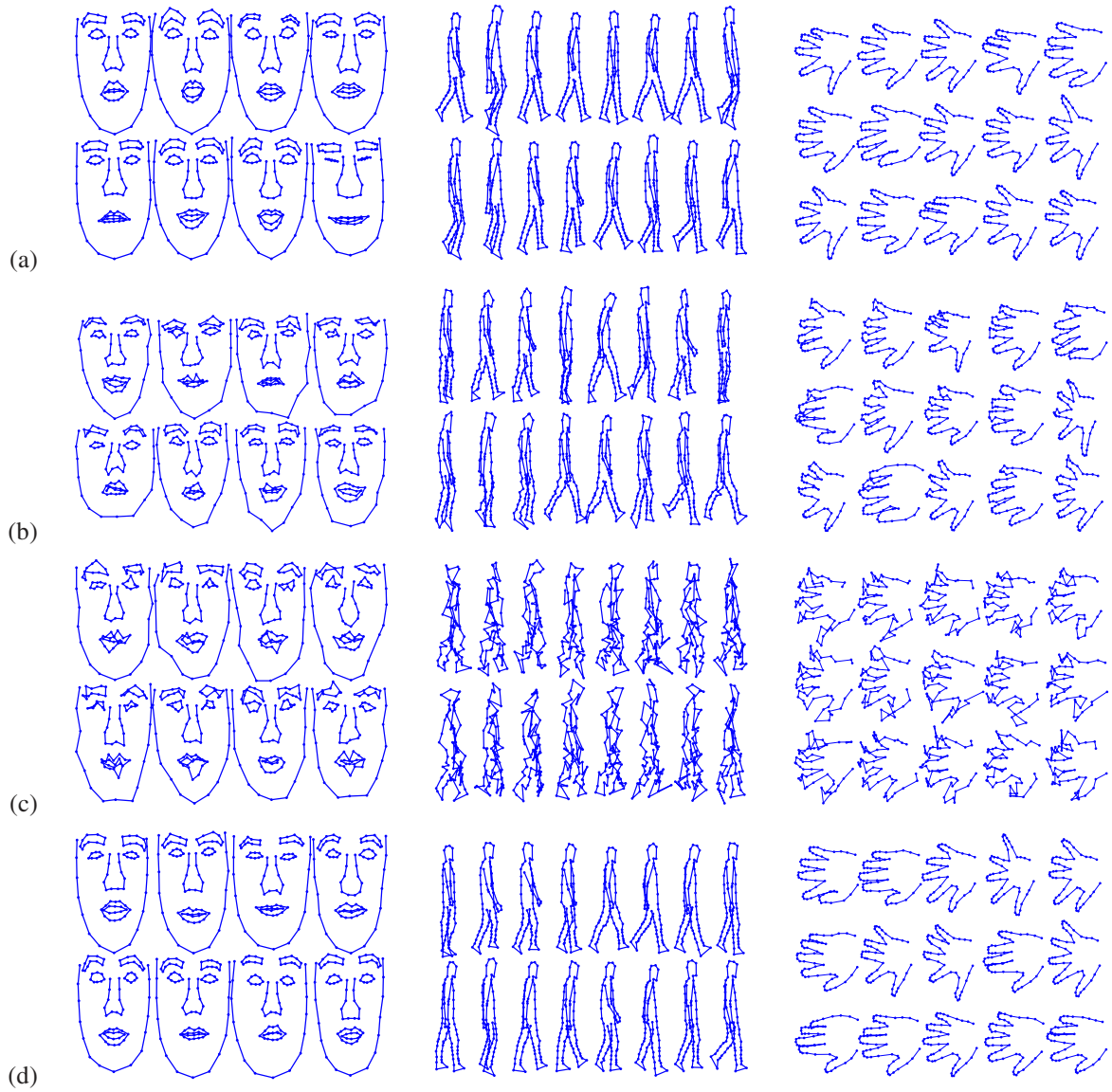

Figure 3. *More samples drawn from a) the graph learned from data, b) the k-fan graph, c) the fully disconnected graph, d) the full graph. These samples empirically justify the representation power of each graph. We observe that although the learned graph (a) is significantly sparser than (b) and (d), it captures the geometrical regularities and spatial constrains of object parts as well as the full graph.*

Structure	face	eye	eyebrow	nose	mouth	silhouette
Learned Graph	2.6	1.6	3.3	1.9	2.4	3.6
K-fan Graph (k=12)	4.0	1.8	4.7	2.2	3.6	6.3

Table 1. Comparing facial component localization errors of k-fan graph and the graph learned data. The table shows the 85% trimmed mean of the average distance (in pixel) between located components and manually labelled ground truth.

[9] A. Martinez and R. Benavente. The ar face database. *CVC Technical Report 24*, June 1998.

[10] N. Meinshausen and P. Buhlmann. High dimensional graphs and variable selection with the lasso. *Annals of Statistics*, 2005.

[11] H. Schneiderman and T. Kanade. A statistical method for 3d detection applied to faces and cars. In *Proceedings of Computer Vision and Pattern Recognition*, 2000.

[12] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal Royal Statististics*, 58(1):267–288, 1996.