3D-GMNet: Single-View 3D Shape Recovery as A Gaussian Mixture

Kohei Yamashita[1]
kyamashita@vision.ist.i.kyoto-u.ac.jp
Shohei Nobuhara[1,2]
nob@i.kyoto-u.ac.jp
Ko Nishino[1]
kon@i.kyoto-u.ac.jp

[1] Kyoto University
Kyoto, Japan

[2] JST, PRESTO
Saitama, Japan

Abstract

In this paper, we introduce 3D-GMNet, a deep neural network for 3D object shape reconstruction from a single image. As the name suggests, 3D-GMNet recovers 3D shape as a Gaussian mixture. In contrast to voxels, point clouds, or meshes, a Gaussian mixture representation provides an analytical expression with a small memory footprint while accurately representing the target 3D shape. At the same time, it offers a number of additional advantages including instant pose estimation and controllable level-of-detail reconstruction, while also enabling interpretation as a point cloud, volume, and a mesh model. We train 3D-GMNet end-to-end with single input images and corresponding 3D models by introducing two novel loss functions, a 3D Gaussian mixture loss and a 2D multi-view loss, which collectively enable accurate shape reconstruction as kernel density estimation. We thoroughly evaluate the effectiveness of 3D-GMNet with synthetic and real images of objects. The results show accurate reconstruction with a compact representation that also realizes novel applications of single-image 3D reconstruction.

1 Introduction

Single-view 3D shape recovery finds many applications in a wide range of domains including robotics, mixed reality, and graphics. Image-based 3D reconstruction is, however, a fundamentally ill-posed problem due to the inherent loss of dimensionality through image projection. Past methods have leveraged constraints arising from projective geometry [1, 26, 27, 35], radiometric surface properties [2, 3, 12], and optical imaging properties [8, 20] to arrive at unique 3D reconstructions. For the even more underconstrained single-view 3D shape recovery, recent works have shown that convolutional neural networks (ConvNet) can be trained end-to-end to impose effective priors for accurate reconstruction [6, 7, 14, 17, 31].

Past methods on single-view 3D shape reconstruction have chiefly employed conventional representations of geometry: point clouds, volumes, and mesh models. Each of these representations have their pros and cons. Point clouds are simple enough to learn their mapping from single images [7, 14, 17], but they lack topological (surface) information. Volumes are straightforward to train and infer with 3D ConvNets [6, 34]. Their resolution is,

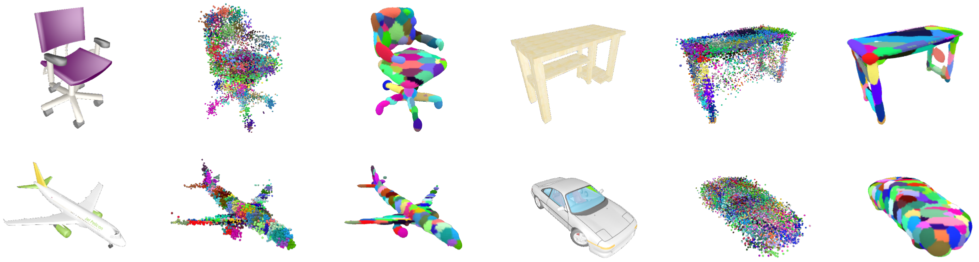

Figure 1: We introduce 3D Gaussian Mixture Network (3D-GMNet) for single-view 3D image-based reconstruction. From a single image, the network recovers object shape as a 3D Gaussian mixture which is extremely compact, integrates properties of conventional geometry presentations, and offers additional benefits for novel applications. In this figure, the object shapes are represented with 256 Gaussians.

however, limited in practice due to the inherent cubic memory cost. Although, in contrast, meshes efficiently represent object surfaces [7, 15], the non-parametric 2D representation makes constraining and reconstructing occluded sides of general objects challenging.

In this paper, we derive a novel method for recovering 3D shape from a single image as a Gaussian mixture. As depicted in Fig. 1, our key contribution lies in enabling the reconstruction of the whole 3D geometry of an object from its single-view observation as a compact, analytical model that can be sampled as a point cloud, interpreted as a volume, and distilled into a surface. We train a ConvNet to learn this mapping from an image to a Gaussian mixture end-to-end by formulating 3D shape recovery as kernel density estimation. The 3D Gaussian mixture shape representation significantly reduces memory footprint compared to volume-based occupancy estimation approaches [6, 34] while providing a straightforward means for defining the surface unlike unstructured point cloud representations [7, 14]. Also in sharp contrast to mesh-based shape representations, the Gaussian mixture model enables the network to adaptively refine the shape topology.

Two recent works have demonstrated the advantages of representing 3D geometry as a Gaussian mixture, purely as a generator [13] or from an input 3D mesh model [9]. Most closely related to our work, Genova *et al*. also demonstrated its use for single-view 3D shape recovery through network distillation [9]. Their representation is, however, inherently constrained to consist of axis-aligned 3D Gaussians, which fundamentally limits the ability to approximate general objects that can have angled structures. Our method is not limited to axis-aligned Gaussian mixtures, hence not bound to carefully axis-aligned objects. Furthermore, our 3D shape recovery is achieved in the viewer-centric coordinate system, *i.e*., the output shape is in the camera coordinate frame, which greatly expands the general applicability of the method, and also enables applications such as pose estimation.

We derive a deep neural network which we refer to as the 3D-GMNet that learns to output a set of parameters of a Gaussian mixture shape model that explains the input image and associated 3D model at training time. We propose two novel loss functions to train 3D-GMNet end-to-end. The first is the 3D Gaussian mixture loss, which evaluates the accuracy of the estimated Gaussian mixture shape model with regards to the target 3D shape. This is achieved by maximizing the likelihood of the Gaussian mixture which in turn is evaluated by considering the target 3D points as samples from the true distribution. The second is the 2D multi-view loss that evaluates the accuracy of the 2D projections of the Gaussian mixture to random viewpoints against the true silhouettes, *i.e.*, the projections of the ground truth 3D shape to the same viewpoints. We show that these 3D and 2D losses work hand-in-hand to estimate accurate and effective Gaussian mixtures for general objects.

We conduct extensive experimental validation of the effectiveness of 3D-GMNet using images of both synthetic and real objects. We also demonstrate the advantages of the estimated 3D Gaussian mixture shape model over conventional geometry representations both qualitatively and quantitatively. Most important, we show that the reconstructed shape model is compact as it only requires the 3D mean and covariance for each mixture component. In addition, it admits a number of direct favorable applications, including controlled level-of-detail reconstruction via Gaussian mixture reduction, pose estimation, and distance measurement. The results show that 3D-GMNet achieves accurate single-image shape estimation with a representation that opens a new avenue of applications of image-based geometry reconstruction.

2 Related Work

Shape Representation Learning-based 3D shape estimation studies can be categorized by their shape representations: voxels [6, 34], point clouds [7, 14, 19], patches [10], mesh models [15, 31], primitive sets [9, 13, 21, 23, 29], and learned functions [18, 22]. Wu *et al.* [34] discretize the target 3D shape into a 128×128×128 voxel grid and their neural network estimates the occupancy of each voxel. This is a memory-intensive approach, although it can handle 3D shapes of different topology in a unified manner. Lin *et al.* [17] propose a network that estimates multi-view depth-maps from a single image. Groueix *et al.* [10] represents the target 3D shape by a collection of 3D patches. Although memory efficient, fusing multiple depth-maps or multiple patches into a single watertight 3D shape remains challenging. Mesh-based approaches [15, 31] can make use of local connectivity of the 3D shape. Handling different topologies, however, becomes an inherently challenging task with meshes. Primitive-based approaches [21, 23, 29] represent the target 3D shape as a collection of simple objects such as cuboids or superquadrics. They can realize a compact representation of the target volume, but cannot represent smooth and fine structures by definition. Mescheder *et al.* [18] train the network as a nonlinear function representing the occupancy probability of 3D object shape. Although highly scalable in resolution, it is a computation-intensive approach since the network should infer the probability for each and every sample. Saito *et al.* [25] use Pixel-aligned Implicit Function to improve memory efficiency albeit specifically for human body shape recovery. Unlike these methods that recover a sampled volume of an implicit function, we recover the parameters of an analytical implicit function, which is much more memory and computation efficient.

Our Gaussian mixture-based representation has the advantages of these conventional shape representations. It models not only the surface points but also the interior of the volume, with an efficient parameterization, *i.e.*, a set of Gaussian parameters, and can generate a watertight 3D surface of arbitrary resolution as its isosurface. It can be considered as a probability density approach with Gaussian distributions, and also as a primitive-based approach with Gaussians as primitives. Additionally, in contrast to cuboid-based approaches, our shape representation can realize 3D registration with a simple canonical algorithm.

Neural Networks for Mixture Density Estimation Mixture density network [4] is a method to predict a target multimodal distribution as a mixture density distribution. Bishop [4] introduced this network architecture with isotropic Gaussian basis functions. Williams [32]

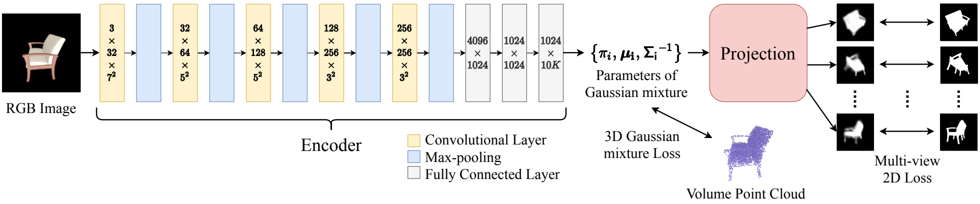

Figure 2: 3D-GMNet is trained end-to-end by finding a Gaussian mixture that best represents the volume point cloud associated with the input image while also minimizing the discrepancy in their multi-view projections.

extended it to utilize a general multivariate Gaussian distribution as the basis function. As described in Sec. 3.2, our density estimation network is inspired by these works.

3 3D Gaussian Mixture Network

Figure 2 shows an overview of our 3D Gaussian Mixture Network (3D-GMNet). Given a 2D image of an object, our 3D-GMNet estimates a set of parameters that defines a Gaussian mixture that best represents the 3D shape of the object in the input image.

3.1 3D Shape as A Gaussian Mixture

Our key idea is to consider the target 3D volume as a collection of observations of a random variable with a Gaussian mixture distribution. Suppose a ground-truth 3D shape is given as a volumetric 3D point cloud. We assume that this 3D point cloud samples the object volume, which can easily be computed from the 3D models from the training data. As such, we may regard them as voxels, too. Each one of the voxels is a sample of the random variable, and our goal when training the network is to estimate a 3D Gaussian mixture distribution that describes these samples best.

A 3D Gaussian mixture distribution is defined as

$$f_{\mathrm{GM}}(\boldsymbol{x}) = \sum_{i=1}^{K} \pi_i \phi(\boldsymbol{x}|\boldsymbol{\mu}_i, \Sigma_i), \tag{1}$$

where K is the number of mixture components and $\{\pi_i\}$ are the mixing coefficients that sum to 1, and each component $\phi(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma)$ is a 3D Gaussian with mean $\boldsymbol{\mu}$ and covariance Σ. Note that the covariance matrix is not limited to be diagonal. Gaussian mixtures can approximate various kinds of distributions with an appropriate K. 3D-GMNet is trained to output the parameters of f_{GM} from a single input image.

Once the density function $f_{\mathrm{GM}}(\boldsymbol{x})$ is obtained, in addition to sampling a 3D point cloud or viewing it as a volume, we can extract the 3D object surface. Assume that we knew the volume of the object V, though it is not available in reality. The object surface is given as the isosurface of the density at $\tau = c/V$, where c decides the level of thresholding. We approximate the unknown $1/V$ by the expectation of the density that can be computed analytically in closed-form

$$\mathrm{E}[f_{\mathrm{GM}}(\boldsymbol{x})] = \int f_{\mathrm{GM}}(\boldsymbol{x})^2 \mathrm{d}\boldsymbol{x} = \sum_{i=1}^{K} \sum_{j=1}^{K} \pi_i \pi_j \phi(\boldsymbol{x} = \boldsymbol{\mu}_i | \boldsymbol{\mu}_j, \Sigma_i + \Sigma_j), \tag{2}$$

since $\mathrm{E}[f_{\mathrm{GM}}(\boldsymbol{x})] = \frac{1}{V}$ holds if $f_{\mathrm{GM}}(\boldsymbol{x})$ is identical to the true distribution $f(\boldsymbol{x})$. The parameter c is determined experimentally in the evaluations in Sec. 4. By thresholding the target space with this value, we obtain the volumetric representation of the 3D shape, which can then be converted to a surface model using the marching cubes algorithm [16].

3.2 Network Architecture

3D-GMNet outputs a set of parameters of a Gaussian mixture $\{\pi_i, \boldsymbol{\mu}_i, \Sigma_i\}_i^K$. As depicted in Fig. 2, the network has an encoder module to predict these parameters and a projection module to render multi-view 2D silhouettes. The encoder consists of 5 convolutional layers, 5 max pooling layers of kernel size 2, and 3 fully-connected layers. Each of the convolution layers is followed by a batch normalization layer and a leaky ReLU activation layer. Each fully-connected layer except the last one is followed also by a leaky ReLU layer.

After these layers, we introduce an output layer tailored for Gaussian mixture parameters to enforce constraints to make it a valid probability density function [4, 32]. The mean $\{\boldsymbol{\mu}_i\}$ should be a 3D position in Euclidean space $\boldsymbol{\mu}_i \in \mathbb{R}^3$, and we use an identity mapping for $\boldsymbol{\mu}_i$ as $\boldsymbol{\mu}_i = \boldsymbol{a}_{\boldsymbol{\mu}_i}$, where $\boldsymbol{a}_{\boldsymbol{\mu}_i}$ is the corresponding output of the last layer. To ensure that the coefficient $\{\pi_i\}$ sum to 1, the output layer applies softmax activation.

The precision matrix Σ_i^{-1} of a Gaussian component should be a symmetric positive definite matrix, and can be decomposed as $\Sigma_i^{-1} = LL^\top$ using Cholesky decomposition where L is a lower triangular matrix. Thus our network predicts $L = \{l_{ij}\}$ instead of Σ_i^{-1} where

$$l_{ij} = \begin{cases} a_{l_{ij}} & i > j\,, \\ \exp(a_{l_{ij}}) & i = j\,, \\ 0 & \text{otherwise}\,, \end{cases} \tag{3}$$

where $a_{l_{ij}}$ is the corresponding output of the last layer. Notice that this enforces l_{jj} to be positive so that the mapping from L to Σ_i^{-1} is bijective.

3.3 3D Gaussian Mixture Loss

We provide 3D shape supervision at training time as voxels. To quantitatively evaluate the fit of the estimated Gaussian mixture to these voxels, we use the Kullback-Leibler (KL) divergence, which amounts to minimizing the cross entropy $-\int p(x)\log q(x)\mathrm{d}x$. By considering the target voxels as observations from the true distribution, we can compute this efficiently with Monte Carlo sampling

$$L_{\mathrm{3D}} = -\frac{1}{|P|}\sum_{\boldsymbol{x}\in P} \log f_{\mathrm{GM}}(\boldsymbol{x})\,, \tag{4}$$

where $f_{\mathrm{GM}}(\boldsymbol{x})$ is the output of our network, $\boldsymbol{x} \in P$ is a sample from the target density $f(\boldsymbol{x})$ and $|P|$ is the number of sampled points. For training, we randomly sample a fixed number of 3D voxels from the original target voxels for each mini batch.

We also introduce a loss that encourages Gaussian components to be distributed within a distance T from the object center

$$L_{\mathrm{dist}} = \frac{1}{K}\sum_{i=1}^{K} \{\mathrm{ReLU}(|\boldsymbol{\mu}_i| - T)\}^2\,. \tag{5}$$

In our experiments, we use $T = 0.85$ to cover the entire object space.

3.4 2D Multi-view Loss

The 3D loss is not sufficient to recover accurate geometric shape, especially for general objects that can have thin, angled structures. For this, we also leverage 2D projections of the 3D shape and derive a differentiable 2D multi-view loss that evaluates the consistency of object silhouettes.

To generate a silhouette of a 3D Gaussian mixture of Eq. (1), we use para-perspective projection[11] for each mixture component since it projects a 3D Gaussian as a 2D Gaussian. As a result, we obtain a 2D Gaussian mixture as a projection of our 3D Gaussian mixture shape representation. Note that perspective projection does not result in a Gaussian due to its nonlinearity. We can derive the para-perspective projection of a 3D Gaussian mixture (see supplementary material for details)

$$d(\boldsymbol{x}) = \sum_{i=1}^{K} \pi_i \phi_{2D}(\boldsymbol{x}|\boldsymbol{\mu}'_i, \Sigma'_i)\,, \tag{6}$$

where $\phi_{2D}(\cdot)$ denotes a 2D Gaussian of the form

$$\phi_{2D}(\boldsymbol{x}|\boldsymbol{\mu}', \Sigma') = \frac{1}{2\pi|\Sigma'|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} g(\boldsymbol{x}|\boldsymbol{\mu}', \Sigma')\right)\,, \tag{7}$$

$$g(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma) = (\boldsymbol{x}-\boldsymbol{\mu})^{\top} \Sigma^{-1} (\boldsymbol{x}-\boldsymbol{\mu})\,. \tag{8}$$

This para-perspective projection is differentiable and denoted as the projection module in Fig. 2. Thanks to this analytical expression of the 2D projection we can directly evaluate the discrepancy with the projection of the predicted shape, unlike methods that rely on 2D kernel density estimation of projected point clouds [19].

We generate a pseudo soft silhouette $\hat{s}(\boldsymbol{x})$ from Eq. (6) to evaluate the consistency of the projected 2D Gaussian mixture with the ground-truth silhouette $s(\boldsymbol{x}) \in [0,1]$. Given a random sampling of Q points from the 2D probability density function $d(\boldsymbol{x})$ of Eq. (6), the probability of observing at least a point out of the Q points at a pixel position $\boldsymbol{x}$ is given by

$$\hat{s}(\boldsymbol{x}) = 1 - \{1 - d(\boldsymbol{x})\}^{Q}\,. \tag{9}$$

By approximating the silhouette generated from the probability density function by this $\hat{s}(\boldsymbol{x})$, we can define an L2 loss

$$L_{\text{sil}}(\hat{s}(\boldsymbol{x}), s(\boldsymbol{x})) = \sum_{\boldsymbol{x}} \{\hat{s}(\boldsymbol{x}) - s(\boldsymbol{x})\}^2\,, \tag{10}$$

as our silhouette loss. In the experiments, we determined Q using validation data. In training, we use 4 random viewpoints to evaluate the 2D multi-view loss.

4 Experimental Results

We first describe data and metrics used for our experiments and then detail quantitative evaluation on synthetic and real images. In addition, we demonstrate 3D pose alignment and automatic level-of-detail shape recovery using 3D-GMNet.

Data For quantitative evaluation, we use 3D models in ShapeNet[5]. Each 3D model in ShapeNet has a polygon CAD model and its volume data. For each polygon model, multi-view RGB images are rendered from random 100 viewpoints at a unit distance from the model using a tessellated icosahedron. To normalize the apparent size of the model in the rendered images, the distance is adjusted on a per-model basis as the diagonal size of the object bounding box. The virtual camera is configured as 128×128 resolution and 68° field-of-view. We train and evaluate our network for 4 object categories, namely *Chair*, *Car*, *Airplane* and *Table*. For real images, we evaluate our method using real chair images in Pix3D Dataset[28]. We remove images in which the object is partly in the image or occluded by other objects for simplicity. Occlusion handling will be addressed in future work. We resize and crop images using manually annotated 2D masks.

Following [28], we use three metrics for evaluation: intersection of union (IoU), earth mover's distance (EMD), and chamfer distance (CD). IoU evaluates the coverage of the estimated volume w.r.t. the ground truth volume, using voxelized Gaussian mixture. Higher IoU means better reconstruction results. EMD and CD evaluate geodesic and shortest distances between two surfaces via point clouds sampled on them, respectively. As described in [28], we uniformly sampled points on the estimated and the ground truth surface to generate a dense point cloud, and then randomly sampled 1024 points from the point cloud. They are scaled to fit a unit cube for normalization for EMD and CD calculation. We used the implementation by Sun *et al.* [28].

Training Parameters We use the Adam optimizer with learning rate of 10^{-4}. The mini batch size is set to 64. Training loss is averaged in each mini batch. We use 80% of the 3D models in ShapeNet for training, 10% for validation, and the rest for testing.

4.1 Single-Image 3D Reconstruction

Fig. 3 shows predicted 3D models. Given the single input image 3D-GMNet estimates the object shape as a 3D Gaussian mixture, which is rendered from two novel views as a mesh model. The renderings from the novel views demonstrate qualitatively that the proposed 3D-GMNet can estimate the full 3D shape including thin, angled structures accurately.

Contribution of 2D Multi-View Loss Table 1(a) shows shape reconstruction accuracy using only the 3D Gaussian mixture loss (3D) and also with the 2D multi-view loss (MV). Fig. 4 shows silhouettes of reconstructed 3D shapes with and without the 2D multi-view loss.

	3D	3D+MV
CD	0.0866	**0.0842**
EMD	0.0923	**0.0889**
IoU	0.466	**0.482**

(a)

	3D-R2N2 [6]	PSGN [7]	3D-VAE-GAN [33]	DRC [30]	Marr Net [34]	Atlas Net [10]	Pix3D [28]	Ours
CD	0.239	0.200	0.182	0.160	0.144	0.125	**0.119**	0.130
EMD	0.211	0.216	0.176	0.144	0.136	0.128	**0.120**	0.129
IoU	0.136	N/A	0.171	0.265	0.231	N/A	**0.287**	0.259

(b)

Table 1: (a) Reconstruction accuracy only using the 3D Gaussian mixture loss (3D) and also with the 2D multi-view loss (3D+MV). The 2D multi-view loss increases reconstruction accuracy. (b) Accuracy of single image 3D reconstructions using real images in Pix3D dataset as reported in [28]. 3D-GMNet (Ours) achieves accuracy comparable to state-of-the-art but with significantly smaller memory footprint and flexible representation.

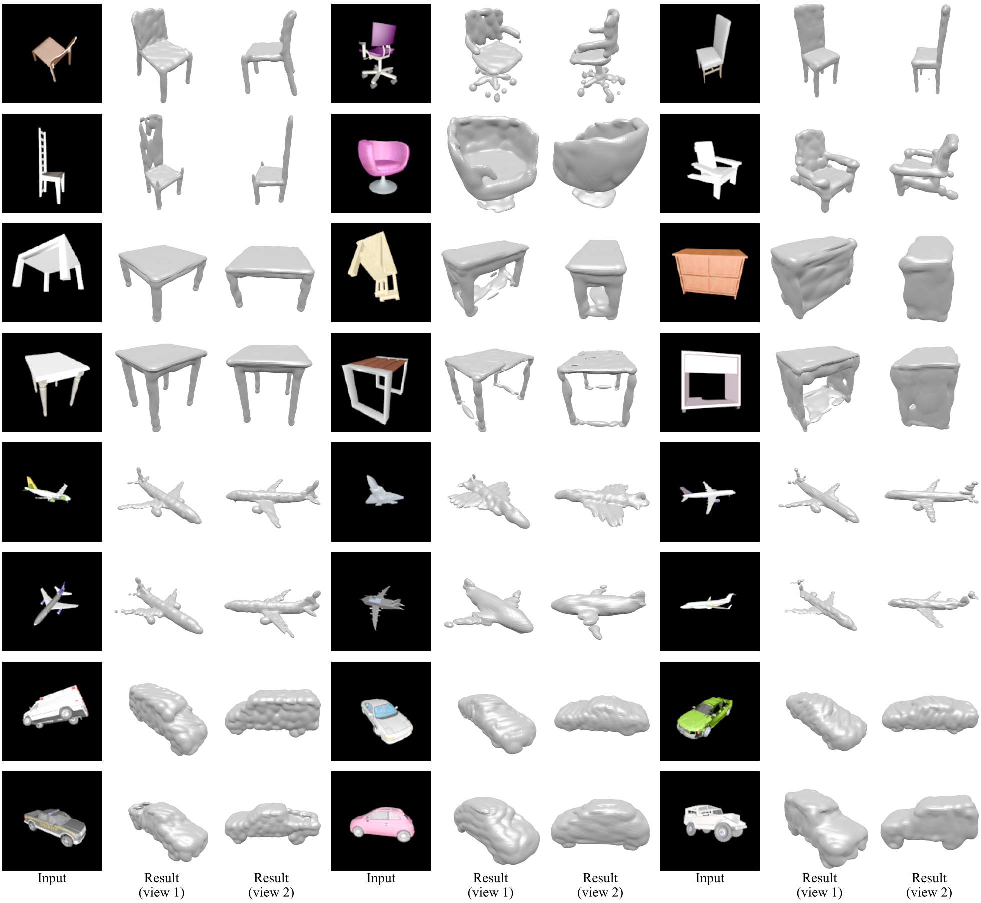

Figure 3: Class-specific single-image 3D reconstruction for 4 object categories with $K = 256$ Gaussian mixture shown as surface meshes from two novel views. 3D-GMNet accurately recovers complex shape including angled and thin structures.

These results clearly show that the silhouette loss reduces 3D shape reconstruction errors and enables recovery of complex geometric structures.

Number of Gaussian Components Fig. 5(a) shows recovery 3D Gaussian-mixture shape models using 3D-GMNet with different numbers of mixture components K. We can observe that though reconstructions with higher K results in a detailed reconstruction, those with lower K also approximates the 3D shape accurately.

Comparison to Genova *et al.* Fig. 6(a) shows qualitative comparison of reconstructed 3D shape of a chair from a single using 3D-GMNet and from its mesh model as shown in Genova *et al*. Our reconstruction is not limited to axis-aligned Gaussians, which results in superior reconstruction even from a single image.

3D Reconstruction from Real Images Fig. 6(b) and Table 1(b) show single-image 3D reconstruction results with real images in the Pix3D dataset[28]. Note that the training scheme (trained for a single category or multiple categories, in object-centered manner or

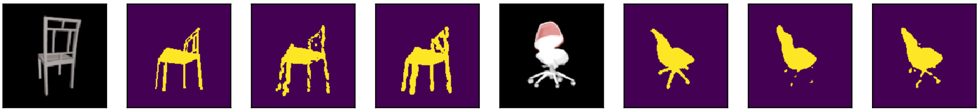

Figure 4: Effectiveness of 2D multi-view loss (from left to right, input image, ground truth silhouette, silhouette of reconstruction without and with the 2D multi-view loss). The 2D multi-view loss is essential for recovering complex structures such as thin chair legs.

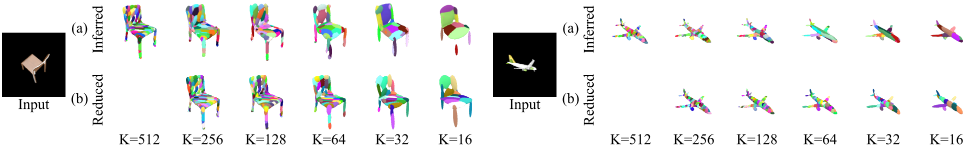

Figure 5: (a) 3D shapes estimated with different numbers of Gaussian components. Different color indicates a distinct Gaussian component. (b) Controlled level-of-detail reconstruction. Results of Gaussian mixture reduction from $K = 512$ to $K = 16, 32, \ldots, 256$. The results show that we can control the level-of-detail of the reconstruction without altering the network while maintaining accuracy.

viewer-centered manner) differs for each method. For this, the quantitative comparison is not necessarily fair. In addition to the several advantages of our shape representation, the reconstruction accuracy of 3D-GMNet is comparable to the state-of-the-art methods.

4.2 3D Pose Estimation

3D-GMNet recovers the shape in the local camera coordinate system of the input image. Given two images of a single object from different viewpoints, 3D-GMNet can recover the 3D object shape in two different coordinate systems, which means that we can estimate the relative pose of the cameras by aligning the estimated 3D shapes. The key challenge for achieving this pose estimation is the view-dependent assignment of Gaussian components in the recovered 3D Gaussian mixture. We solve this by aligning the covariance matrices of Gaussian mixtures from different viewpoints. In the supplemental material, we show that this can be computed analytically. Fig. 7 shows the alignment results. The results show that our method can provide reasonable pose alignments without explicit point cloud generation.

4.3 Controlled Level-of-Detail Reconstruction

Fig. 5(b) demonstrates controlling the level-of-detail of shape reconstruction by automatically varying the number of components of the Gaussian mixture shape model. Given an input image (the leftmost column), our network with $K = 512$ infers a Gaussian mixture representation of its 3D shape (the second column). By applying Gaussian mixture reduction[24], we can obtain different level-of-details of the underlying 3D shape as shown in the second and the fourth rows. When compared with the 3D shapes estimated by 3D-GMNet originally trained with the corresponding number of components (the first and the third rows), the controlled level-of-detail reconstruction yield similar accuracy.

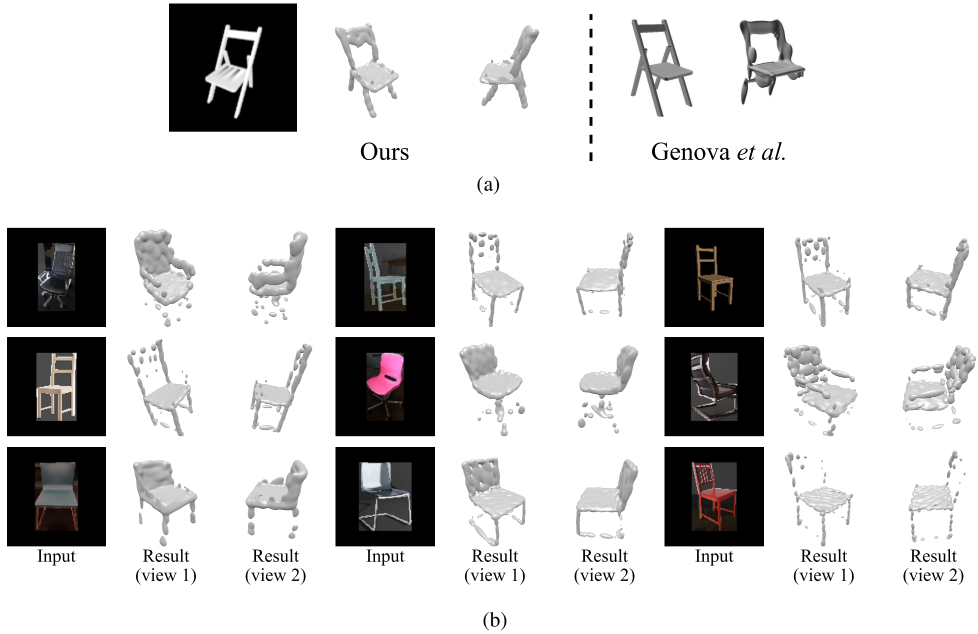

Figure 6: (a) Left : Input image and output isosurface of our method. Right : Input mesh and reconstruction result reported in Genova *et al.* [9]. (b) Reconstructions from real images in the Pix3D dataset[28] shown as surface mesh models rendered from two novel views.

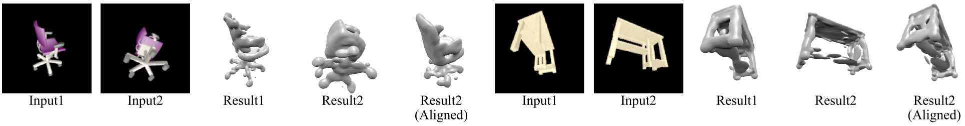

Figure 7: 3D pose estimation results. 3D-GMNet enables estimation of relative camera pose between input images (input2 to input1).

5 Conclusion

We proposed 3D-GMNet for recovering the 3D shape of an object from its single-view observation as a Gaussian mixture. We introduced a 3D Gaussian mixture loss and a 2D multiview loss to accurate reconstruct the 3D shape from a single image. Experimental results show that our 3D-GMNet successfully estimates the object 3D shape as a compact Gaussian mixture that can be sampled and viewed as conventional geometry representations including point cloud, volume, and mesh model. Extensive experimental validation showed that the method can recover 3D shape accurately even with a lower number of components, while maintaining comparable performance with state-of-the-art methods, but with the additional benefits of this unique shape representation including pose estimation and level-of-detail control.

Acknowledgement This work was in part supported by JSPS KAKENHI 17K20143 and JST PRESTO Grant Number JPMJPR1858.

References

[1] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building Rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

[2] Amit Agrawal, Ramesh Raskar, and Rama Chellappa. What is the range of surface reconstructions from a gradient field? In *Proc. ECCV*, pages 578–591. Springer, 2006.

[3] Neil Alldrin, Todd Zickler, and David Kriegman. Photometric stereo with non-parametric and spatially-varying reflectance. In *Proc. CVPR*, pages 1–8. IEEE, 2008.

[4] Christopher M. Bishop. Mixture density networks. Technical report, Aston University, 1994.

[5] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An information-rich 3d model repository. *arXiv:1512.03012*, 2015.

[6] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Proc. ECCV*, 2016.

[7] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proc. CVPR*, 2017.

[8] Paolo Favaro, Stefano Soatto, Martin Burger, and Stanley J Osher. Shape from defocus via diffusion. *TPAMI*, 30(3):518–531, 2008.

[9] Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T. Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *Proc. ICCV*, 2019.

[10] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proc. CVPR*, 2018.

[11] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003. ISBN 0521540518.

[12] Carlos Hernández, George Vogiatzis, Gabriel J Brostow, Bjorn Stenger, and Roberto Cipolla. Non-rigid photometric stereo with colored lights. In *Proc. ICCV*, pages 1–8, 2007.

[13] Amir Hertz, Rana Hanocka, Raja Giryes, and Daniel Cohen-Or. PointGMM: a neural GMM network for point clouds. *arXiv:2003.13326*, 2020.

[14] Li Jiang, Shaoshuai Shi, Xiaojuan Qi, and Jiaya Jia. Gal: Geometric adversarial loss for single-view 3d-object reconstruction. In *Proc. ECCV*, 2018.

[15] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *Proc. CVPR*, 2018.

[16] Thomas Lewiner, Hélio Lopes, Antônio Wilson Vieira, and Geovan Tavares. Efficient implementation of marching cubes' cases with topological guarantees. *Journal of graphics tools*, 8(2):1–15, 2003.

[17] Chen-Hsuan Lin, Chen Kong, and Simon Lucey. Learning efficient point cloud generation for dense 3d object reconstruction. In *Proc. AAAI*, 2018.

[18] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proc. CVPR*, 2019.

[19] K. L. Navaneet, Priyanka Mandikal, Mayank Agarwal, and R Venkatesh Babu. CAPNet: Continuous approximation projection for 3d point cloud reconstruction using 2d supervision. In *Proc. AAAI*, volume 33, pages 8819–8826, 2019.

[20] Shree K. Nayar and Yasuo Nakagawa. Shape from focus. *TPAMI*, 16(8):824–831, 1994.

[21] Chengjie Niu, Jun Li, and Kai Xu. Im2struct: Recovering 3d shape structure from a single rgb image. In *Proc. CVPR*, 2018.

[22] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proc. CVPR*, 2019.

[23] Despoina Paschalidou, Ali Osman Ulusoy, and Andreas Geiger. Superquadrics revisited: Learning 3d shape parsing beyond cuboids. In *Proc. CVPR*, 2019.

[24] Andrew R. Runnalls. Kullback-Leibler approach to gaussian mixture reduction. *IEEE Transactions on Aerospace and Electronic Systems*, 43(3):989–999, 2007.

[25] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proc. ICCV*, 2019.

[26] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1-3):7–42, 2002.

[27] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proc. CVPR*, pages 4104–4113, 2016.

[28] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3D: Dataset and methods for single-image 3d shape modeling. In *Proc. CVPR*, 2018.

[29] Shubham Tulsiani, Hao Su, Leonidas J. Guibas, Alexei A. Efros, and Jitendra Malik. Learning shape abstractions by assembling volumetric primitives. In *Proc. CVPR*, 2017.

[30] Shubham Tulsiani, Tinghui Zhou, Alexei A. Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *Proc. CVPR*, 2017.

[31] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2Mesh: Generating 3d mesh models from single rgb images. In *Proc. ECCV*, 2018.

[32] Peter Williams. Using neural networks to model conditional multivariate densities. *Neural computation*, 8:843–54, 06 1996.

[33] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Proc. NIPS*, pages 82–90, 2016.

[34] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, William T Freeman, and Joshua B Tenenbaum. MarrNet: 3d shape reconstruction via 2.5d sketches. In *Proc. NIPS*, 2017.

[35] Onur Özyeşil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion. *Acta Numerica*, 26:305–364, 2017.