

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

Kohei Yamashita¹
kyamashita@vision.ist.i.kyoto-u.ac.jp

Shohei Nobuhara^{1,2}
nob@i.kyoto-u.ac.jp

Ko Nishino¹
kon@i.kyoto-u.ac.jp

¹ Kyoto University
Kyoto, Japan

² JST, PRESTO
Saitama, Japan

1 Para-perspective Projection of a 3D Gaussian Mixture

To generate multi-view silhouettes of our 3D Gaussian mixture shape representation, we use para-perspective projection[[1](#)] for each mixture component. We derive para-perspective projection of a 3D Gaussian mixture from a different viewpoint.

Since we are only interested in object shape recovery, we can safely assume that the camera pose with respect to the object is defined by rotation about its center. Suppose we project a 3D Gaussian mixture defined in the world coordinate system to a camera of pose R . A 3D Gaussian rotated by R is given by

$$\begin{aligned} f_{\text{GM}}(\mathbf{x}; R) &= \sum_{i=1}^K \pi_i \phi(\mathbf{x} | R\boldsymbol{\mu}_i, R\Sigma_i R^T), \\ &= \sum_{i=1}^K \pi_i \phi(\mathbf{x} | \boldsymbol{\mu}'_i, \Sigma'_i), \end{aligned} \tag{1}$$

where R is the rotation matrix transforming from the world coordinate system to the camera local coordinate system.

Figure 1 shows an overview of para-perspective projection. Para-perspective projection first defines a 3D plane Π_i located at the centroid of the object and parallel to the image plane. Since we project each of Gaussian component independently, the centroid is identical to the mean of each Gaussian $\boldsymbol{\mu}'_i = R\boldsymbol{\mu}_i$.

Suppose an oblique coordinate system centered at $\boldsymbol{\mu}'_i = R\boldsymbol{\mu}_i$ and whose x and y axes are identical to the original Euclidean system but its third w axis is the direction from the camera center (*i.e.*, the origin of the camera coordinate system) to the centroid. Transforming a 3D point \mathbf{x} in the camera coordinate system to a 3D point $\mathbf{y} = (u, v, w)^T$ in this oblique system

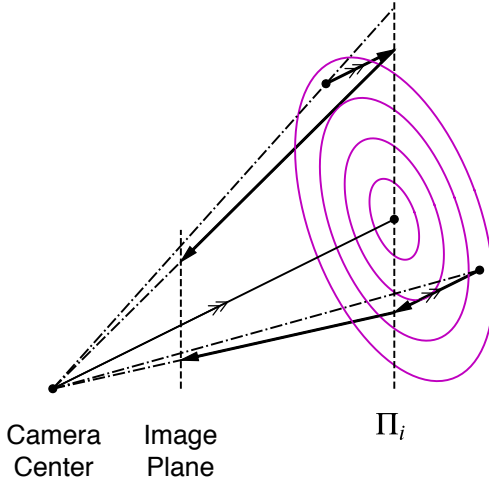


Figure 1: Para-perspective projection of a 3D Gaussian yields a 2D Gaussian on the image plane.

can then be described by

$$\mathbf{x} = (\mathbf{e}_x^\top, \mathbf{e}_y^\top, \boldsymbol{\mu}_i'^\top) \mathbf{y} = \mathbf{M}_i \mathbf{y} \quad (2)$$

$$\mathbf{y} = \mathbf{M}_i^{-1} \mathbf{x}, \quad (3)$$

where $\mathbf{e}_x = (1, 0, 0)^\top$ and $\mathbf{e}_y = (0, 1, 0)^\top$. Therefore, the Gaussian of Eq. (1) is transformed to a Gaussian of parameters

$$\boldsymbol{\mu}_i'' = \boldsymbol{\mu}_i', \quad \Sigma_i''^{-1} = \mathbf{M}_i^\top \Sigma_i^{-1} \mathbf{M}_i, \quad (4)$$

and the parallel projection to the plane Π_i is given by marginalizing this Gaussian $\phi(\mathbf{x} | \boldsymbol{\mu}_i'', \Sigma_i'')$ in the w direction

$$\boldsymbol{\mu}_i''' = (\mathbf{e}_x^\top, \mathbf{e}_y^\top) \boldsymbol{\mu}_i'', \quad \Sigma_i'''^{-1} = \begin{pmatrix} s_{00} - \frac{s_{02}^2}{s_{22}} & s_{01} - \frac{s_{12}s_{02}}{s_{22}} \\ s_{10} - \frac{s_{12}s_{02}}{s_{22}} & s_{11} - \frac{s_{12}^2}{s_{22}} \end{pmatrix}, \quad (5)$$

where s_{ij} denotes the (i, j) element of $\Sigma_i''^{-1}$. By applying a 2D affine transform from the plane Π_i to the image plane, we obtain the 2D Gaussian on the image plane as

$$\boldsymbol{\mu}_i'''' = \frac{\boldsymbol{\mu}_i'''}{z_i}, \quad \Sigma_i''''^{-1} = \frac{1}{z_i^2} \Sigma_i'''^{-1}, \quad (6)$$

where z_i is the depth of the centroid, *i.e.*, the z element of $\boldsymbol{\mu}_i'$.

As a result, the para-perspective projection of a 3D Gaussian of Eq. (1) is given by

$$d(\mathbf{x}) = \sum_{i=1}^K \pi_i \phi_{2D}(\mathbf{x} | \boldsymbol{\mu}_i''', \Sigma_i'''), \quad (7)$$

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

$$\phi_{2D}(\mathbf{x} | \boldsymbol{\mu}''', \Sigma''') = \frac{1}{2\pi |\Sigma'''|^{\frac{1}{2}}} \exp \left(-\frac{1}{2} g(\mathbf{x} | \boldsymbol{\mu}''', \Sigma''') \right). \quad (8)$$

This para-perspective projection is differentiable and denoted as the projection module in Figure 2 of the paper.

2 Details of 3D Pose Estimation

3D-GMNet recovers the shape in the local camera coordinate system of the input image. Hence we can estimate relative pose of cameras by aligning the estimated 3D shapes. We show this is done by aligning the covariance matrices of Gaussian mixtures analytically.

The covariance matrix of a Gaussian mixture is given by

$$\Sigma_{\text{GM}} = \sum_{i=1}^K \pi_i \left(\Sigma_i + (\boldsymbol{\mu}_i - \boldsymbol{\mu}_{\text{GM}})(\boldsymbol{\mu}_i - \boldsymbol{\mu}_{\text{GM}})^\top \right),$$

where $\boldsymbol{\mu}_{\text{GM}} = \sum_{i=1}^K \pi_i \boldsymbol{\mu}_i$. We then estimate the rotation via diagonalization with its eigenvectors sorted in descending order according to the magnitude of the corresponding eigenvalues. The correspondence based on the order of eigenvalues has a sign ambiguity on each of the eigenvectors. To find the correct signs, we evaluate the rotation that minimizes the $L2$ distance between two Gaussian mixtures:

$$\begin{aligned} \int \left(f_{\text{GM}}^{(1)}(\mathbf{x}) - f_{\text{GM}}^{(2)}(\mathbf{x}) \right)^2 d\mathbf{x} &= \sum_{i,j} \Phi^{(1,1)}(i,j) \\ &+ \sum_{i,j} \Phi^{(2,2)}(i,j) - 2 \sum_{i,j} \Phi^{(1,2)}(i,j), \end{aligned} \quad (9)$$

where

$$\begin{aligned} \Phi^{(a,b)}(i,j) &= \pi_i^{(a)} \pi_j^{(b)} \int \phi_i^{(a)}(\mathbf{x}) \phi_j^{(b)}(\mathbf{x}) d\mathbf{x} \\ &= \pi_i^{(a)} \pi_j^{(b)} \phi \left(\mathbf{x} = \boldsymbol{\mu}^{(a)} | \boldsymbol{\mu}^{(b)}, \Sigma^{(a)} + \Sigma^{(b)} \right). \end{aligned} \quad (10)$$

References

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