Uncertainty Quantification for Incomplete Point Cloud

Debabrata Ghosh

July 24, 2024

Contents

1 Background

In this work, we will attempt to tackle the problem of uncertainty quantification (UQ) of curved surfaces conditioned on a given incomplete point cloud. Point clouds are widely used to represent 3D shapes because of the ease of obtaining point cloud data (via scanning). But often the captured 3D geometry is either noisy and sparse or even incomplete due to limited sensor resolution and occlusion. Modelling 3D shapes using point cloud often requires high sampling density and more accurate registration. Though existing works [Yua+18; Yu+21; Wan+24; Cui+23; Li+23] implement point cloud completion for given noisy or incomplete point clouds by trying to predict the missing points accurately, these methods do not give us the uncertainty of individual points of the proposed completion. Through this work, we will try to address the task of quantifying spatial uncertainty of the points of the missing parts of an incomplete point cloud. A practical benefit of this approach can be found in 3D scanning for active vision or next-best-view planning by quantifying how useful a prospective next sensor position would be. We can use our learnt probability distribution to take the uncertainty into account while reconstructing a surface or repairing a noisy point cloud. We can also compute regional probabilities for collision detection which gives us a probability of colliding to a particular object rather than just a deterministic answer.

[SJ22] and [SJ23] performed the UQ task by using a stochastic version of Poisson Surface Reconstruction (PSR) method. [SJ22] introduced the idea of stochastic PSR by interpreting the observed points along with the normal information in a point cloud as observations of a Gaussian Process (GP). The GP posterior in turn gives us the mean and covariance functions of the vector field defined for a given point cloud with normals. Consequently, the mean and covariance function of the scalar field can be computed from that of the vector field which in turn gives us the uncertainty of the reconstructed surface. Unfortunately, such method is only applicable for oriented point clouds and normal information is often not available for captured 3D data. Rather than reconstructing the surface directly we can also consider the implicit representation of the 3D shape defined by distance fields or distance function over a given space that maps any query point to the distance to its closest point on the surface. [WF07] introduced the idea of GP based implicit surface representation for 3D shapes and [Lee+19; Wu+20] improved upon such methods. [LG+24] implemented the idea of estimating distance fields from noisy point clouds using GP regression. Our aim is to apply the same idea to an incomplete point cloud to get a probabilistic estimation of the distance field and further update the parameters of a Gaussian mixture model or the kernel function of a GP regression model conditioned on the observed complete point cloud using a Neural Network. So, we can then use the learned parameters or kernel and network to estimate the uncertainty of the distance field for a new incomplete point cloud.

2 Task Definition

For a dataset of point clouds $\mathbb{R}^{N_P \times N_C \times 3}$ with N_P instances of N_C points (complete set) in each point cloud instance, assume we observe N_O points and N_M points are missing where $N_O + N_M = N_C$. For a single measurement $\mathbf{X} = \left\{\boldsymbol{x}_i \in \mathbb{R}^3\right\}_{i=1}^{N_C}$ of 3D coordinates, we also append another variable representing surface distances, denoted as $\boldsymbol{Y} = \{y_i\}_{i=1}^{N_C}$, forming the complete dataset for training $(\mathbf{X}, \mathbf{Y})$. Our goal is to learn a combination of models which when given N_O points of a new instance, outputs a probabilistic estimate of the distance field of the corresponding 3D shape. We can then use our output for further downstream tasks. Following are the methods that we plan to implement in order to achieve our goals:

2.1 Baseline: Gaussian Mixture Regression

As a baseline we want to model the distance field for the incomplete point clouds using Gaussian mixture (GM) regression. A standard Gaussian Mixture Model is composed of K Gaussian components, described by a set of parameters $\Theta = \{(\pi_k, \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k})\}_{k=1}^{K}$, as given in:

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(x \mid \boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}\right) \tag{2}$$

where π_k is the mixture weight, $\boldsymbol{\mu_k}$ and $\boldsymbol{\Sigma_k}$ are the mean and covariance of the k-th Gaussian. Given a perfect K, the components will properly approximate the object surface. We estimate the parameters of our GM regression by jointly maximizing the likelihood of the complete data $(\mathbf{X}, \mathbf{Y})$ conditioned on the encoding of the observed points.

Earlier works have also used a Hierarchical tree structure to help with the Gaussian mixture model selection and speed up Expectation-Maximization (EM) with large numbers of components [Eck+16]. We can apply this idea to our GM regression similar to [Her+20].

2.2 Gaussian Process Regression

As discussed in section 1, we use GP regression to predict the distance field for a given incomplete point cloud with N_O points. Once we have such a GP regression model for the distance field, we can use it to interpolate the distance function at the remaining N_M points which we observe from the complete point cloud data. The idea is to learn the kernel function of the GP regression by maximizing the likelihood of the original N_C points in the complete point cloud while conditioning on the encoding of the N_O observed points. We use the pipeline shown in Fig.1 to train our models.

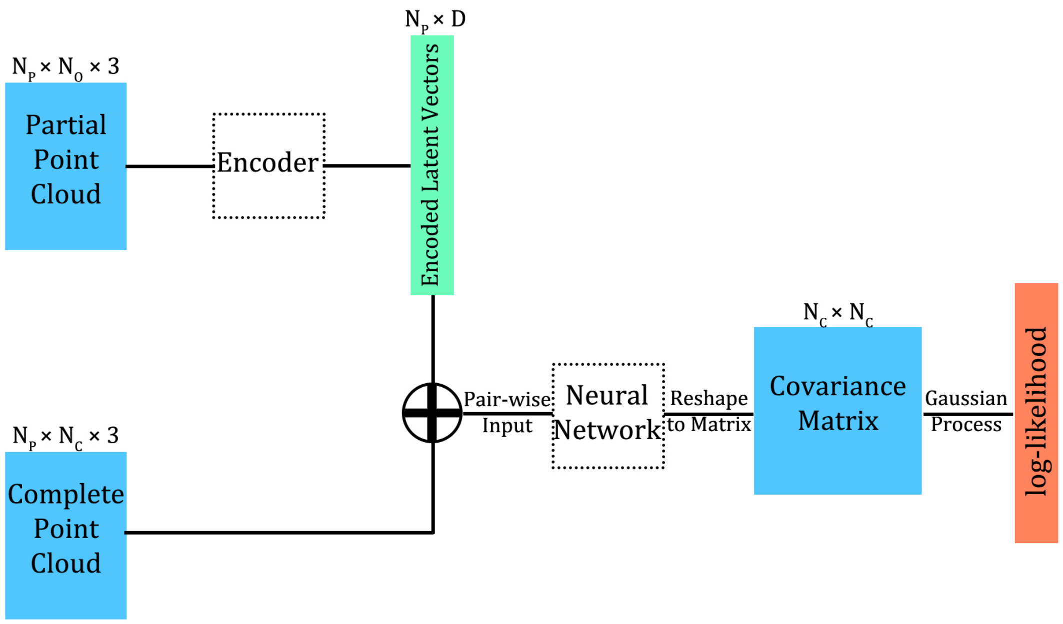

Figure 1: Training pipeline used for learning the kernel function

2.3 Evaluation

To evaluate the accuracy of our learned distance field and reliability of the uncertainty, we propose certain quantitative and qualitative methods described below.

Initially, we'll test our approach by implementing the methods described in section 2 on some hand-picked 2D point cloud data. We will show that with more training points, our model becomes more certain about the shape and the likelihood of points close to the surface is higher compared to distant points. Afterwards, we'll benchmark our baseline model in 2.1 with the GP regression model in 2.2 as well as other existing approaches which provide uncertainty measures such as GPIS [WF07] and log-GPIS [Wu+20]. For accuracy, we can simply compare the distance fields of testing points among these methods by using Root Mean Squared Error (RMSE) between the ground truth and resulting distances. For reliability, we measure the log-likelihood of the points on the surface as it quantifies how well the resulting uncertainty measure agree with the true distribution of the error (higher likelihood implies better agreement). We will also try to see how our method fairs compared to the ones which include the normal information for the point clouds such as in [SJ23]. For qualitative comparison, we will present rendered explicit surfaces from the distance field along with their corresponding uncertainties for the different methods.

3 Schedule

- Implementation of Baseline (GMM based UQ) (6 weeks)

- Stardard Gaussian Mixture Regression (3 weeks)
  - Hierarchical GM Regression (3 weeks)

- Implementation of GP based UQ (8 weeks)
  - Vanilla GP model (1 week)
  - Neural Network based GP model (4 weeks)
  - GP model with GMM prior or other relevant priors (3 weeks)

- Comparison between GMM and GP (3 weeks)
  - Dataset creation (1 week)
  - Quantitative comparison (2 week)
  - Qualitative comparison (simultaneously with quantitative)

- Comparison with Existing Works (2 weeks)
  - Comparison with methods without normal information (1 week)
  - Comparison with methods including normal information such as [SJ23; SJ22] (1 week)

- Thesis Writing (6 weeks)
  - Write the thesis report

References

[Cui+23] Ruikai Cui et al. *P2C: Self-Supervised Point Cloud Completion from Single Partial Clouds.* 2023. arXiv: `2307.14726`.

[Eck+16] B. Eckart et al. "Accelerated Generative Models for 3D Point Cloud Data". In: *Proceedings of (CVPR) Computer Vision and Pattern Recognition.* 2016, pp. 5497 –5505.

[Her+20] Amir Hertz et al. "PointGMM: a Neural GMM Network for Point Clouds". In: *CoRR* abs/2003.13326 (2020). arXiv: `2003.13326`. URL: `https://arxiv.org/abs/2003.13326`.

[LG+24] Cedric Le Gentil et al. "Accurate Gaussian-Process-Based Distance Fields With Applications to Echolocation and Mapping". In: *IEEE Robotics and Automation Letters* 9.2 (2024), pp. 1365–1372. DOI: `10.1109/LRA.2023.3346759`.

[Lee+19] Bhoram Lee et al. "Online Continuous Mapping using Gaussian Process Implicit Surfaces". In: *2019 International Conference on Robotics and Automation (ICRA).* 2019, pp. 6884–6890. DOI: `10.1109/ICRA.2019.8794324`.

[Li+23] Yuhan Li et al. *3DQD: Generalized Deep 3D Shape Prior via Part-Discretized Diffusion Process.* 2023. eprint: `arXiv:2303.10406`.

[SJ23] Silvia Sellán and Alec Jacobson. "Neural Stochastic Screened Poisson Reconstruction". In: *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* (2023).

[SJ22] Silvia Sellán and Alec Jacobson. "Stochastic Poisson Surface Reconstruction". In: *ACM Transactions on Graphics* (2022).

[Wan+24] Jun Wang et al. "PointAttN: You Only Need Attention for Point Cloud Completion". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 38.6 (2024), pp. 5472–5480. DOI: `10.1609/aaai.v38i6.28356`. URL: `https://ojs.aaai.org/index.php/AAAI/article/view/28356`.

[WF07] Oliver Williams and Andrew Fitzgibbon. "Gaussian Process Implicit Surfaces". In: *Gaussian Processes in Practice.* 2007. URL: `https://www.microsoft.com/en-us/research/publication/gaussian-process-implicit-surfaces-2/`.

[Wu+20] Lan Wu et al. "Faithful Euclidean Distance Field from Log-Gaussian Process Implicit Surfaces". In: *arXiv preprint arXiv:2010.11487* (2020).

[Yu+21] Xumin Yu et al. "PoinTr: Diverse Point Cloud Completion with Geometry-Aware Transformers". In: *ICCV*. 2021.

[Yua+18] Wentao Yuan et al. "PCN: Point Completion Network". In: *2018 International Conference on 3D Vision (3DV)*. 2018, pp. 728–737.