Seminar Report
Multiview Neural Surface Reconstruction by Disentangling Geometry and Appearance

Jorge Augusto Calvimontes Robles (7024517)
3D Object Reconstruction
Saarland University

Abstract

This work summarizes the ideas presented on the seminar, considering previous and further models up to the presentation of this paper. An overview of the paper is presented, clarifying and organizing it to ease comprehension. The main contributions of the paper are described along with their results. Finally a discussion of the advantages and flaws of the model is presented, pointing out possible improvements.

1 Introduction

Throughout the seminar several techniques to reconstruct 3D geometry were presented. Each one of them presented their own paradigm and challenges. Techniques such as point processing based on point clouds [1] [2], volumetric grids [3], mesh estimation [4] [5] and signed distance functions [6] [7] where contemplated.

Up to this point in the seminar, the models presented where only concerned with the reconstruction of geometry. This paper serves as a bridge to a new consideration, the appearance of the object. Which depends on the material properties and the lighting conditions of the scene.

Considering the topics presented in the seminar, it was observed that this paper had unique and innovative ideas from which other techniques could benefit from. In the same manner, the model presented on this paper could benefit from other ideas of most recent papers. Under that understanding, the structure of the paper is as follows: first a comparison with previous or related techniques is presented in section 2, then an overview of the model for this paper is described in section 3 clarifying and ordering the main ideas, on section 4 quantitative and qualitative results are discussed, on section 5 possible improvements are considered w.r.t other models and finally on section section 6 the conclusions are presented.

2 Previous Work

Based on the discussions on the seminar, it was observed that SDFs had certain advantages in comparison to previous techniques under certain contexts. A optimal use case, can consider the use of images (given that they are simple to acquire) to produce a detailed 3D reconstruction that has a low memory footprint.

Considering the type of input data, models that can be trained using images have an advantage over point processing, given that input data is easier to obtain. In this sense the model is not dependent on a specialized acquisition method as it is the case for point clouds which are acquired thought a specialized sensor.

In terms on memory usage, an SDF works by calculating the distance from one point to the desired geometry. On the other hand, voxel grids are a direct mapping on a 3D grid space, this can be optimized using an octrees. Even with this optimization, the scability is still limited by the desired output resolution which correlates to the amount of memory used. This is not the case of SDFs which don't rely on mapping. Which allows it to have a more compact representation, leading to memory efficiency and open the possibility for more detailed reconstructions.

Referring to the topology of an object, it was discussed that mesh estimation is limited on their type of topology. That is, an initial shape (ellipsoid) is deformed by adding or changing the position of vertices. Given that the model is the one doing this process, deforming the mesh such that edges are broken or added is not an accountable operation. Given that there is no way that the model

could remember or reorganize the relationship between vertices for such cases. This limits the model to certain type of topologies (genus 0) e.g. shapes without holes. By the other hand, SDFs work on a continuous space, where the distance from a point with a direction to the object is calculated. Which allows it to work with more complex topologies.

When referring to SDFs, there are different approaches to train one. [6], [7] train their models using a mesh and calculating the distance w.r.t one vertex and [8] considers a volumetric grid. Both of these methods depend on 3D supervision.

An alternative to this methods is differentiable rendering, in which the geometry of the object can be inferred based on images. This considers a ray tracing model in which a pixel on an image is correlated to a point in an object, considering appearance properties. In such a case, if the model is capable of generating a similar image as in training it can be said that the model learned the geometry implicitly. It was demonstrated by [9] that an SDF can learn a geometry on an implicit way by employing an Eikonal regularization. [10] uses this technique to learn the geometry implicitly based on silhouettes of images. [11] also used this, but considered an appearance model. On their differentiable volumetric render, a SDF approximates the geometry while the appearance is represented by a texture map.

3 Methodology

The model is composed of two neural networks. An implicit geometry network (IG) with the job of implicitly learning the geometry of the object. By the other hand, a Render Network (RN) in charge of generating an image based on the output and other parameters obtained from the previous model.

The input data that the model uses is composed of multi-view images with their corresponding camera positions and masks. The masks, in this case only account for the object to be reconstructed. The model will approximate the interaction of light modeled by ray tracing. To this end, it will use the sphere tracing algorithm [12] to calculate the intersection distance using an SDF f. Given a camera central point c a ray is casted with a direction v thought certain distance t, until it hits the surface of the object at $\hat{x}$.

Considering the input data, each image and their camera parameters will be used to generate rays. These rays will start from the (given) camera center and travel along a (sampled) direction passing through a corresponding pixel and intersecting a surface point. The intersected pixel of the image will be later used to approximate the appearance at the intersection point. Additionally if a pixel is outside the mask of the image, their corresponding ray will be considered as non intersecting.

3.1 Implicit geometry

This network consist of an SDF and an embedding layer. The SDF is represented by a MLP and the embedding layer by a positional encoding embedding. In order to train the SDF f with training parameters θ to correctly approximate a geometry. A common intersection point $\hat{x}$ for rays on different views must be correctly approximated on a gradient descend manner. To that end, the authors propose to use implicit differentiation on the zero level of an SDF, $f(\hat{x}) \equiv 0$. Which implies differentiating w.r.t v, c, θ and solving the derivatives of t. With that considerations they propose,

$$\begin{aligned} x_0 &= c + t_0 v \\ \hat{x}(\theta, \tau) &= c + t_0 v - \frac{v}{\nabla_x f(x_0; \theta_0) \cdot v_0} f(x_0; \theta) \end{aligned} \quad (1)$$

with τ being the camera parameters and t_0 a sampled distance according to the sphere tracing algorithm. Given that this equation makes use of the gradient w.r.t to x of the SDF and the actual value of the SDF, it can be seen as a minimization problem. In which the parameters of f are updated such that the second term converges to zero, in which case the SDF needs to approximate the correct geometry. It's worth remembering that this operation will be done on multiple views, which imply that parameters will be updated differently.

On a similar manner they represent the normal at the intersection point as:

$$\hat{n}(\theta, \tau) = \nabla_x f(\hat{x}(\theta, \tau); \theta) / ||f(\hat{x}(\theta, \tau); \theta)||_2 \quad (2)$$

3.2 Neural Renderer

On this paper it is assumed that the surface light field radiance L of a material depends on the bidirectional reflectance distribution function (BRDF) and the illumination of the scene. This can be modeled by the rendering equation [13]. Given that the rendering equation is dependent on the direction of a ray v, its intersection point $\hat{x}$ and it's normal $\hat{n}$, the authors propose to approximate the light field using a MLP (M).

$$L(\theta, \gamma, \tau) = M(\hat{x}, \hat{n}, v; \gamma) \quad (3)$$

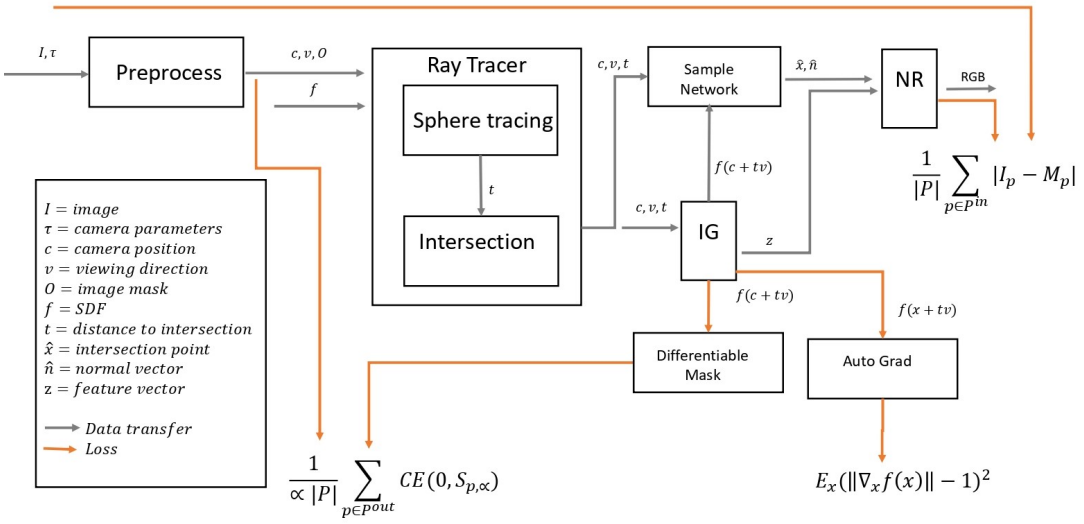

Figure 1: Training pipeline

with γ being the trainable parameters that account for appearance. Given that the input of the network M is dependent of the output of f. The resulting radiance represented as a color, should be approximated to the color of the sampled pixel for the corresponding intersecting ray.

Although M represent the approximation of a continuous light field, it doesn't account for global illumination. To this end, the result of the embedding layer z presented previously is used as input to M. This allows the renderer to reason globally about the geometry.

$$L(\theta, \gamma, \tau) = M(\hat{x}, \hat{n}, v, z; \gamma) \tag{4}$$

3.3 Training

In order for the model to approximate the light field and the implicit geometry it takes into consideration 3 losses. The first loss considers the L2 norm difference between the sampled color pixel from image I and the generated color from the renderer M,

$$l_{RGB} = \frac{1}{|P|} \sum_{p \in P^{in}} |I_p - M_p| \tag{5}$$

with p being the pixel being evaluated and P^{in} the set of all pixels inside the mask.

To account for the pixels outside the mask, an approximation of the differentiation of the binary mask is used,

$$S_\alpha(\theta, \tau) = sigmoid(-\alpha \min_{t \geq 0} f(c + tv; \theta) \tag{6}$$

with α being a hyperparameter. This is then used for the mask loss,

$$l_{mask}(\theta, \tau) = \frac{1}{\alpha |P|} \sum_{p \in P^{out}} CE(0, S_{p,\alpha}(\theta, \tau)) \tag{7}$$

with CE being cross entropy loss, which considers the case where there is no mask (zero). Finally as a regularization term that will help the MLP approximate a SDF an Eikonal term is considered,

$$l_E(\theta) = \mathbb{E}_x(||\nabla_x f(x; \theta)|| - 1)^2 \tag{8}$$

Figure 1 shows an overview of the pipeline, showing which type of data is passed on each stage. Data to calculate the loss is also shown.

4 Experiments and Results

Four types of experiments were conducted: reconstruction with known cameras, few cameras, unknown cameras and disentangling geometry. All present quantitative and qualitative results, except for the later that only has qualitative results. Additionally and ablation study is presented to show the influence of different parameters. Throughout the experiment the DTU MVS dataset [14] is used. This include different scenes composed of multi-view images of different objects, with their corresponding camera annotations.

For the experiment with known cameras a model is trained for each scene using all their images. The Chamfer-L1 distance and the PSNR reconstruction metrics are used to compare the model with other methods like: DVR[11], Colmap[15] and Furu [16]. The quantitative results show that the model

had better reconstruction metrics compared to the other models. This can also be observed on their qualitative results in which it can be observed that the model generates reconstructions with less artifacts and with fine details on the surface. Additionally, upon observing the rendered reconstruction. It can be seen diffuse, glossy and specular components, which serves as an indicator that the neural renderer was able to approximate the appearance of the object in the scene.

On a similar manner, the model was trained with few cameras and compared only with Colmap. The reconstruction, again quantitatively was better; while qualitatively better details, less artifacts and realistic appearance are observed. For the experiment with unknown cameras, the authors relied on the SIFT [17] algorithm to find correspondences between images and then through linear interpolation generate a noisy approximation of the camera parameters. Through this experiment the authors wanted to demonstrate that the model was capable of accounting noisy cameras and correcting them. The results, are similar to the previously discussed.

On their ablation study they train the model without considering certain parameters and optimizations. In terms of reconstruction details, it is observed that the viewing direction, the normal and the feature vector have a great impact. Without these parameters lighting and geometry can be confused which leads to artifacts. Additionally if noisy camera parameters are used without accounting for their correction, very bad reconstruction are obtained. This highlights the importance of using accurate camera parameters or optimizing noisy data correctly.

The model trains two neural networks, one that accounts for geometry (IG) and another for appearance (RN). Given that the normal for an intersection point is not computed by the network, rather calculated on an intermediate stage. The authors claim this condition allows for disentanglement of geometry and appearance. Which means, training on different scenes with different appearances and then just changing the RN between geometries, should be possible. This claim is proven to be true, by looking at the qualitative results of transferred appearance. It can be observed that the material properties adapt to the new geometry and also their specular, diffuse and glossy properties.

5 Discussion and Improvements

During the presentation of this paper, important observations were discussed. The most prominent ones highlighted the dependence of the model on camera parameters and masks. This dependence limits the model to only work under lab settings in which the acquisition of the input relies on specialized equipment and pre-processing.

For camera parameters, the authors recognized this limitation, suggesting the use other methods to calculate the camera coordinates based on multi view images. In the paper, SIFT is used to find correspondences between multi-view images. Considering this approach already achieves good results, an improvement could be done by having a better feature detector. This could be done using two paradigms: model based models and neural networks. For the former, good alternatives to SIFT can be SURF [18] and KAZE [19]. These two methods have the advantage of having better matching precision. Additionally, SURF is the fastest among them.

Another alternative is the use of neural networks, using convolutional neural networks has been demonstrated by [20] [21] to have better descriptors than SIFT. A suggested improvement can consist of either pre-training a model to calculate camera parameters or to jointly train the model. The later will entail, extending the Implicit Geometry network by adding convolutional layers. This idea may be plausible given that the authors already extended the SDF with a positional encoding embedding. A further improvement can consider directly the flow field between images instead of the camera parameters. For that end, FlowNet 2.0 [22] could be used.

Regarding the use of masks, the authors mention that for each scene they had to manually annotate each of the masks, which is a problem if the model want to be used outside a lab environment. It was also pointed out during discussion, that the use of mask give a strong clue to the model about the geometry. As it was shown by [11] an SDF is capable of learning the geometry based on the silhouette from different views. This can also be the case for this model, only that the learning is done implicitly.

In order to eliminate the dependence of masks. A suggested improvement can consist of either pre-training a model to do the segmentation or to jointly train the model considering a masking module. This again will signify extending the Implicit Geometry network, in this case to account for seg-

mentation. To this end Equation 7 could be replaced with a corresponding loss for such a model. Although this idea still need to be proven, a good candidate could be Masked-attention Mask Transformer [23] which has proven to be a robust and efficient method for segmenting images.

Referring to the neural renderer, it was trained to approximate the surface radiance field of different objects. These objects were composed of materials which presented a continuous BRDF. On this regard two observations were made, the model is not capable of disentangling lighting from material properties and complex materials modeled with non continuous functions are not accounted.

For the first case, in order to factor lighting and material, two networks could be trained with their corresponding losses. For the lighting network this could be estimated using an architecture similar to [24]. This model works with stereo vision, given that the model uses multi view images it is possible to implement it taking this consideration. As for the estimation of the BRDF, in order to also tackle the second observation it would be convenient to use a Space Varying BRDF (SVBRDF) which is a non continuous function and as such more complex to estimate, allowing to render more realistic images. In order to approximate this function [25], [26] and [27] have demonstrated it is possible using either deep or convolutional neural network. Finally, the rendering network will be feed using the output of both networks and considering the normal and intersection point. A good candidate for this type of network will be the differentiable renderer of [28] which approximates the light radiance as a sum of diffuse and a specular lobes over 24 Spherical Gaussians.

Considering the above, it would be interesting to observe a disentanglement between lighting and SVBRDF and try it for different geometries. On that topic, the disentanglement presented on this paper have many potential use applications on fields that involve design of products e.g. clothes and furniture design. It can also be used on augmented reality. If such improvements can be applied, the model has the potential to be flexible and work on an outside environment.

6 Conclusions

Throughout the seminar many 3D reconstruction methods were observed, each one with their own advantages depending on the context. At the time of presenting this paper, SDFs were the main topic and as such a comparison with previous methods was done. This paper served as a bridge between only geometry reconstruction and reconstruction considering appearance.

The method for reconstructing 3D geometries using an implicit geometry network and neural renderer was presented. Making clarification that were not explained in detail on the paper. Then an interpretation of the quantitative and qualitative results was discussed. Finally the main points discussed during the presentation of the paper were presented. And for such possible alternatives were suggested.

This paper present very innovative ideas. Highlighting: the use of two specialized networks, calculating the geometry through implicit differentiation of the intersection point, accounting for global illumination using a positional embedding as a feature vector, calculating the normal independent of the network as to allow disentanglement and transfer of appearance. Considering the ideas of papers which estimate the appearance using another paradigm. New ideas emerged which could help the previously mentioned.

References

[1] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.

[2] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[3] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs," in *Proceedings of the IEEE international conference on computer vision*, pp. 2088–2096, 2017.

[4] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2mesh: Generating 3d mesh models from single rgb images," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 52–67, 2018.

[5] S. Liu, T. Li, W. Chen, and H. Li, "Soft rasterizer: A differentiable renderer for image-based 3d reasoning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7708–7717, 2019.

[6] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 165–174, 2019.

[7] R. Chabra, J. E. Lenssen, E. Ilg, T. Schmidt, J. Straub, S. Lovegrove, and R. Newcombe, "Deep local shapes: Learning local sdf priors for detailed 3d reconstruction," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pp. 608–625, Springer, 2020.

[8] Y. Jiang, D. Ji, Z. Han, and M. Zwicker, "Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization (supplementary materials),"

[9] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman, "Implicit geometric regularization for learning shapes," *arXiv preprint arXiv:2002.10099*, 2020.

[10] Y. Liao, K. Schwarz, L. Mescheder, and A. Geiger, "Towards unsupervised learning of generative models for 3d controllable image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5871–5880, 2020.

[11] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3504–3515, 2020.

[12] J. C. Hart, "Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces," *The Visual Computer*, vol. 12, no. 10, pp. 527–545, 1996.

[13] J. T. Kajiya, "The rendering equation," in *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pp. 143–150, 1986.

[14] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs, "Large scale multi-view stereopsis evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 406–413, 2014.

[15] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys, "Pixelwise view selection for unstructured multi-view stereo," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pp. 501–518, Springer, 2016.

[16] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 8, pp. 1362–1376, 2009.

[17] D. G. Low, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, 2004.

[18] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[19] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "Kaze features," in *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, pp. 214–227, Springer, 2012.

[20] P. Fischer, A. Dosovitskiy, and T. Brox, "Descriptor matching with convolutional neural networks: a comparison to sift," *arXiv preprint arXiv:1405.5769*, 2014.

[21] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks (2015)," *arXiv preprint arXiv:1406.6909*.

[22] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2462–2470, 2017.

[23] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pp. 1290–1299, 2022.

[24] P. P. Srinivasan, B. Mildenhall, M. Tancik, J. T. Barron, R. Tucker, and N. Snavely, "Lighthouse: Predicting lighting volumes for spatially-coherent illumination," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8080–8089, 2020.

[25] M. Boss, V. Jampani, K. Kim, H. Lensch, and J. Kautz, "Two-shot spatially-varying brdf and shape estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3982–3991, 2020.

[26] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau, "Single-image svbrdf capture with a rendering-aware deep network," *ACM Transactions on Graphics (ToG)*, vol. 37, no. 4, pp. 1–15, 2018.

[27] Z. Li and N. Snavely, "Cgintrinsics: Better intrinsic image decomposition through physically-based rendering," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 371–387, 2018.

[28] M. Boss, R. Braun, V. Jampani, J. T. Barron, C. Liu, and H. Lensch, "Nerd: Neural reflectance decomposition from image collections," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12684–12694, 2021.