Single-Image 3DGS Scene Reconstruction with Geometry-Aware Priors

Machine Visual Perception Course Project Report

November 26, 2025

Information

Authors: TEST
Group Number: TEST

1 Chapter 1: Introduction and Motivation

1.1 Section 1.1: Introduction to the problem

[Provide a thorough introduction to the problem and why it is important. Briefly explain what general techniques there are and how your project fits.]
RCONSTRUCTING 3D MODELS FROM IMAGES IS ACHANLEGNING TASK THAT REQUIRES LARGES NUMBER OF REFERNCE/INPTU IMAGES AND INTENSE PROCESSING TYPICALLY DONE WITH ENRF OR OTHER MODERN ADVANCEMENTS FOCUS ON REDUCING INPOUT PHOTO NUMBER ND PROCESSIGN COMPELXITY SPECIFICALLY, Developments in 3D Gaussian Splatting methods allow for 3D scene reconstruction using single or few RGB images. While faster than other scene reconstruction techniques and requiring only a "one-shot" pass, these approaches often suffer from challenges such as layout/scale drift, over-smooth geometry and hallucinations in occluded regions. This project focuses on one recent method, SplatterImage, as a baseline. By storing 3D Gaussians in an image, it reduces reconstruction to learning an image-to-image neural network, allowing the use of a 2D U-Net to form the representation. Each pixel stores the parameters for a corresponding 3D Gaussian, allowing for reconstruction in a single feed-forward pass. This reduces the need for large amounts of compute. Despite its speed, SplatterImage does have some issues that have been noted in related works, particularly in reconstructing structures unseen in the input view, including for views significantly different from the source. This project aims to address this by proposing a lightweight augmentation to the model, by integrating explicit geometry priors (such as planes, normals, visibility cues, depth, segmentation, edge maps) with minimal architectural changes.
    EXPAND/REWORD ABOVE

1.2 Section 1.2: Background and related work

[Include a few very relevant related works and how your work relates to those, expanding on the previous section. We do not expect you to cover all previous works.] discuss 3dgs discuss triplane? discuss splatterimage

1.3 Section 1.3: Overview of the idea

[Provide an overview stating why the idea of the project makes sense and what the main motivation is.] splatteriamge suffers from hallucination and problems simply due to lack of data feeding models additional data improves reconstruction with modern compute and ml advancements there now exist many good quality pretrained geometry related models for example generating depth, normal maps, segmentation we propose using the knowledge/capacibiltiy of these models to predict additional priors of input images, creating a

modified model that accepts these priors these should result in an improved recostruction quality we propose performing ablation study to see which priors are most effective/significant in changing the reconstruction quality

2   Chapter 2: Method

2.1   Section 2.1: Baseline algorithm

[Explain the baseline architecture you used to build your algorithm on. You may reproduce figures from the original papers.]

explain 3dshg output explain spaltteriamge architecture with diagram

2.2   Section 2.2: Algorithm improvements

[Explain what you implemented to improve over the baseline. You may include figures to explain the idea and logic. Focus on the ideas and not the implementation.]

explain insertion of additional layers explain addition of transofrmer/FiLM layers to allow multimodal input from segmentation tokens explain addition of LORA matrices due to compute limitations/allowign working of exisitng model

2.3   Section 2.3: Implementation details

[Explain how you implemented the improvements. You may include code snippets with the corresponding explanations.]

2.3.1   Normal Map Exploration

We studied using normal maps as a 3D spatial prior to augment the visual tokens for the Splatter Image. A normal map stores surface normal data as RGB colour information, which shows how light interacts with the surface at a per-pixel level, hence we wanted to investigate if that could help the model generate the 3D surface of the input image.

We investigated using various different models to generate a normal map from the input image.

For our ground truths we use [1] which presents a Deep Implicit Surface Network to generate a 3D mesh from a 2D image by predicting the underlying signed distance fields and provides a dataset of higher resolution images of the ShapeNet models from [2] each paired with a depth image, a normal map and an albedo image at `https://github.com/Xharlie/ShapenetRender_more_variation`. We then feed these images into the normal map generation models and compare them to the given normal maps to evaluate their performance.

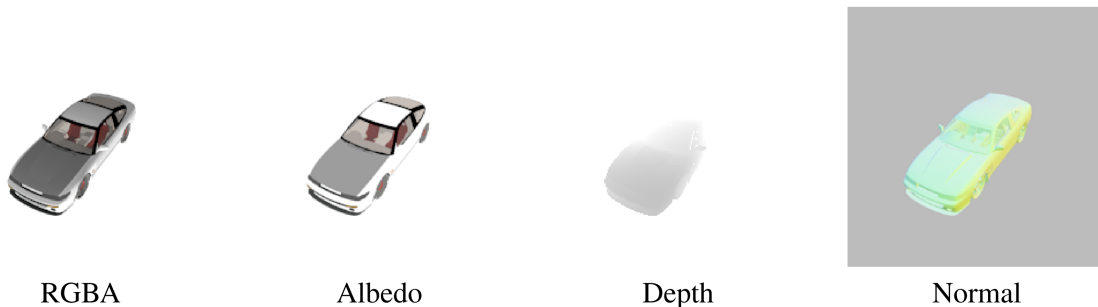RGBA   Albedo   Depth   Normal

Figure 1: Example of image with maps used as ground truth taken from [3]

For the models We referenced [3] which implements a network which estimates the per-pixel surface normal probability distribution and uses uncertainty-guided sampling to improve the quality of prediction of surface normals. The paper provided code at `https://github.com/baegwangbin/surface_normal_uncertainty` that implemented this method on a network trained on NYUv2 [4], with the ground truth and data split

provided by GeoNet [5] [6] and another trained on ScanNet [7], with the ground truth and data split provided by FrameNet [8]. Both models take in the input image and dimensions and return a normal map of the input image with the same dimensions as output.

We run both pretrained models on the dataset.

<div align="center">

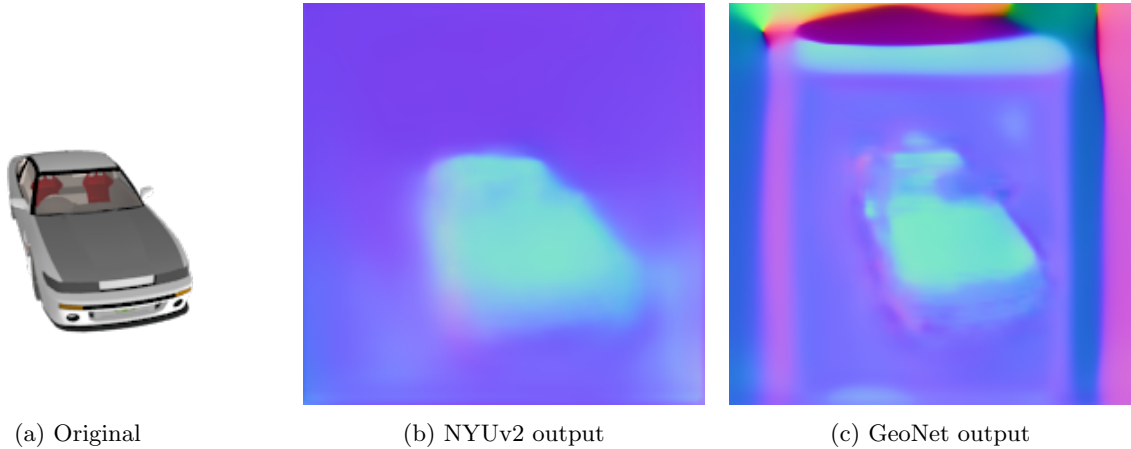(a) Original   (b) NYUv2 output   (c) GeoNet output

Figure 2: Comparison of original input and two model outputs

</div>

We then try passing in larger input dimensions than the actual ones into the models, such that a normal map larger than the original input is produced. We then resize the image to the original input dimensions.

<div align="center">

(a) NYUv2 output   (b) GeoNet output   (c) Ground truth from [1]
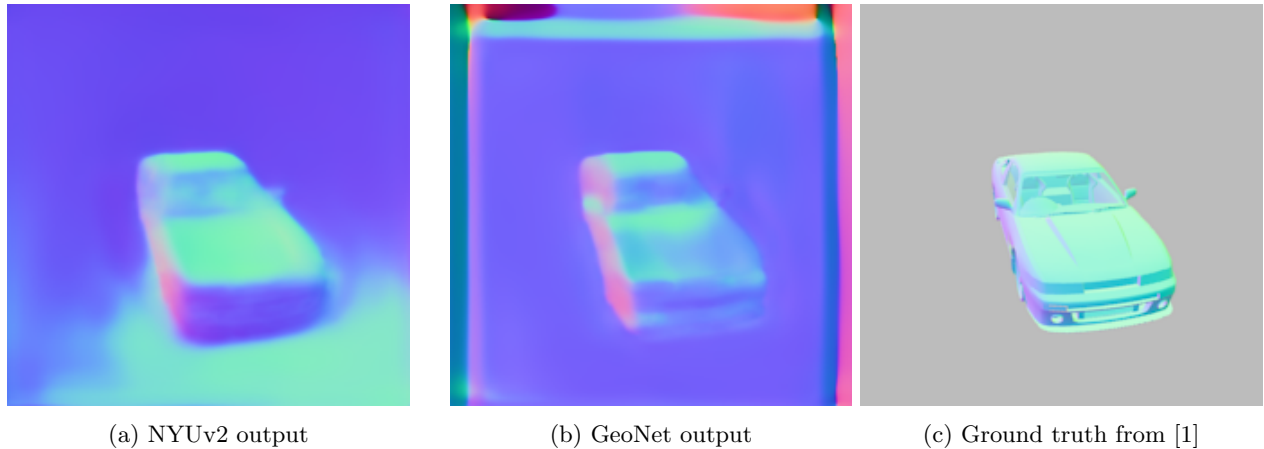
</div>

Figure 3: Comparison of model outputs with input dimensions 448x448 instead of 224x224 alongside ground truth

The normal map generated for images when given larger input dimensions seem to have more clearly defined edges and surface contouring. Details on the evaluation of the two different models (trained on NYUv2 and GeoNet respectively) are in section 2.5.

(talk about final model chosen and show example on a shapenet model)

2.3.2 Depth Map Exploration

Depth maps store the distance of a surface from the camera per-pixel. These distances vary in type, such as metric, which considers the physical distance from the camera to the observed point, and relative (such as those produced by the models below). Monocular depth estimation (MDE) models input just a singular image, and produce a depth map (relative distance).

Produced depth maps were compared against the "ground truths" produced by `https://github.com/Xharlie/ShapenetRender_more_variation`, as was done in the normal priors exploration. An example of the depth map produced by them is visible in Figure **??**. However, it is important to note that these depth map "ground truths" were not always perfect, as can be seen in the following example:

- (INSERT) Image needed here of poor ground truth.

This inclined us to take the quantitative results produced by comparing MDE models tested against these ground truths with a pinch of salt. For each produced depth map, the following metrics were used to compare against the ground truths.

1. **Absolute Relative Error**: Measures the average difference between the predicted depth and the ground truth, normalised by the ground truth depth.

2. **Root Mean Squared Error (RMSE)**: Calculates the standard deviation of the residual errors.

3. **Scale-invariant RMSE (SI-RMSE)**: Computes the RMSE while ignoring the unknown absolute scale and shift between the prediction and ground truth.

4. δ at **1.25** ($\delta_{1.25}$): Represents the percentage of predicted pixels p that satisfy the condition $\max(\frac{p}{p^{gt}}, \frac{p^{gt}}{p}) <$ 1.25, which takes into account close pixel-wise agreement.

The following table (LINK to label) summarises the mean metrics across the MiDaS models tested.

2.3.3 Prior Model Integration

TWO MODEL SCRIPTS, PREGENERATED FOR TRAINING/EVAL/VAL FINAL MODAL ORCHESTRATOR PATTERN PER MODEL SCRIPT BACTEHD, SAVES TO IMAGE DUE TO SPACE LIMITATIONS GENERATE PARQUET PER PRIOR PARQUETS UPLOADED TO HF

ON THE FLY VERSION? ANOTEHR OCRHSESTOROR, IMAGES ADDED AND FED INTO MODEL TO ALLOW NOVEL IMAGES

2.3.4 Model Changes

MODEL CHANGES added layer parameters on creation if congfig, inject lora and perform weight graft

2.3.5 Training and Evaluation Changes

new priors stacked into input eval/train on ready dataset have minimal changes added new dataloader code novel evaluator?

2.4 Section 2.3: Data pipelines

[Explain your data format, how you consume the data in your algorithms, and data augmentation.]

EXPLAIN SRN CARS format EXPLAIN TRANSFORMATION TO PARQUET EXPLAIN STORE OF uint8s UPLOADED TO HF

TRAINIGN LOADS HF DATASET LOAD HF PRETRAINED WEIGHTS PERFORMS WEIGHT GRAFT TRAINING OCCURS WEIGHTS FUSED UPLOADED TO HF

2.5 Section 2.4: Training procedures

[Explain which framework and optimizers you use, how you implemented the training logic.]

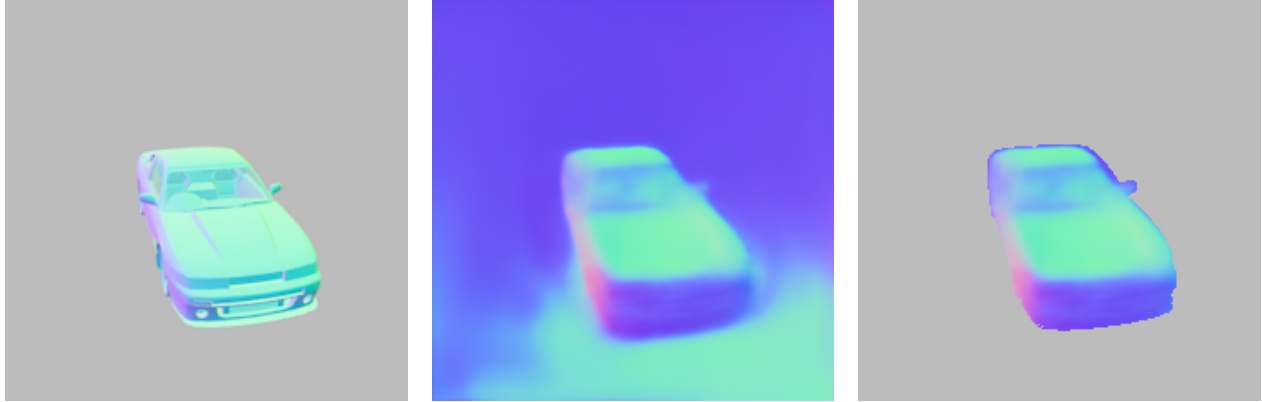USE TORCH LIST LORA INSERTTION LIRBARY LIST BASELINE WEIGHT PARAMS

WE WRITE TRAINIG NNTOEBOOK RUSN ON CLOAB WITH A100 LSIT TRAINING PARAMS WEIGHTS FUSED AND UPLAODED TO HF

2.6 Section 2.5: Testing and validation procedures

[Explain which framework you use, how you implemented the testing/ validation logic.]

2.6.1 Normal map priors evaluation

To compare our generated normal maps to the ground truth normal maps provided in [1], we first mask out the background of the generated normal maps such that the difference in background colour does not contribute to the evaluation metrics for normal map generation.

(a) Ground truth from [1]   (b) NYUv2 output (448x448) version   (c) Output with background masking

Figure 4: Example of masking out background for model evaluation against ground truth

We then use Visual Information Fidelity to compare the normal maps generated by the two models to the ground truth.

3 Chapter 3: Experiments and Evaluation

3.1 Section 3.1: Datasets

[Explain the datasets utilized: what they contain, why they are utilized, assumptions, limitations, possible extensions.]

3.2 Section 3.2: Training and testing results

[Explain the training and testing results with graphs and elaborating on why they make sense, what could be improved.]

3.3 Section 3.3: Qualitative results

[Show in figures and explain visual results. Include different interesting cases covering different aspects/ limitations/ dataset diversity. If not converged, explain what we can expect once converged. Include any other didactic examples here.]

3.4 [Optional] Section 3.4: Quantitative results

[A table and associated explanations for quantitative results.]

3.5 [Optional] Section 3.5: Comparison to state-of-the-art

[Qualitative and/ or quantitative comparisons to one or more recent works, especially the baseline work.]

4 Chapter 4: Conclusions and Future Directions

4.1 Section 4.1: Conclusions

[Summarize what the project was about and the main conclusions.]

4.2 Section 4.2: Discussion of limitations

[Explain the limitations of your technique. You may want to refer to previous sections or show figures on the limitations.]

halluciantion in hidden areas still a problem data, lot of data and compute needed

4.3 Section 4.3: Future directions

[State a few future directions for research and development. These typically follow from the discussion on limitations.]

References

[1] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In *NeurIPS*, 2019.

[2] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing 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. *CoRR*, abs/1512.03012, 2015.

[3] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *International Conference on Computer Vision (ICCV)*, 2021.

[4] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[5] Xiaojuan Qi, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia. Geonet: Geometric neural network for joint depth and surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 283–291, 2018.

[6] Xiaojuan Qi, Zhengzhe Liu, Renjie Liao, Philip HS Torr, Raquel Urtasun, and Jiaya Jia. Geonet++: Iterative geometric neural network with edge-aware refinement for joint depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[7] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

[8] Jingwei Huang, Yichao Zhou, Thomas Funkhouser, and Leonidas Guibas. Framenet: Learning local canonical frames of 3d surfaces from a single rgb image. *arXiv preprint arXiv:1903.12305*, 2019.