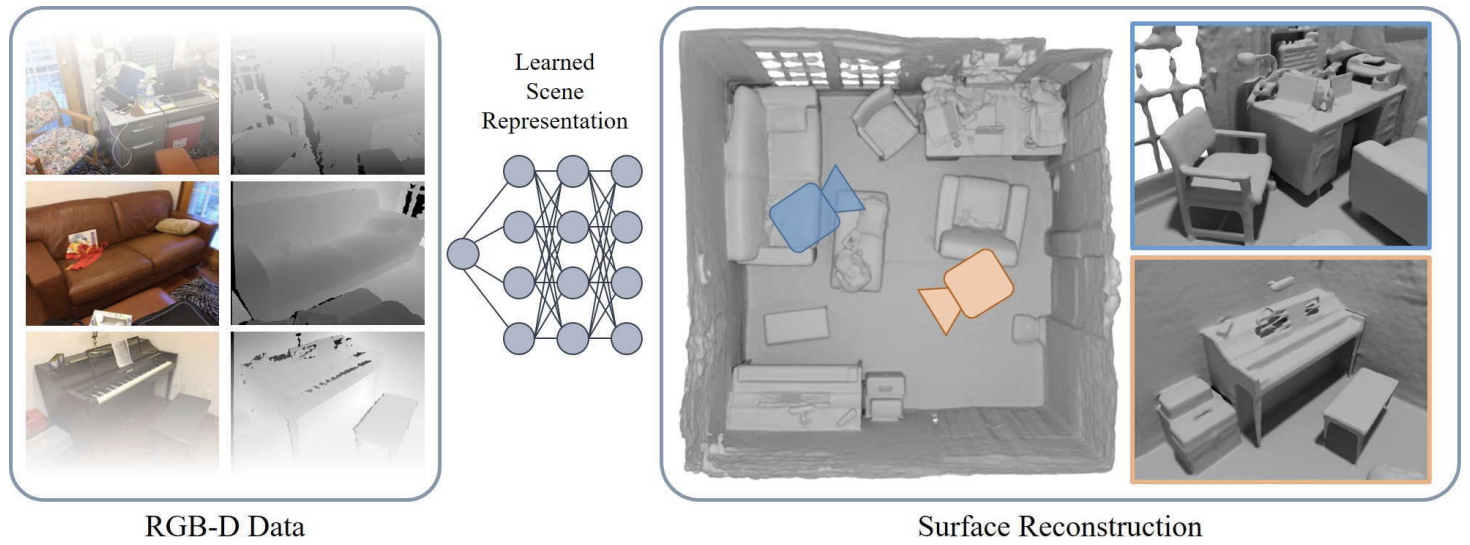
Neural RGBD



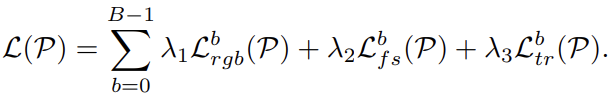
Neural RGBD uses SDF-modified version of neural radiance field to apply NeRF network in scene reconstruction tasks instead of rendering images synthesis. The input data of this model is a series of RGB-D images along with camera intrinsic and poses of each captured image. Then, the input data are fed in a multilayer perceptron that learns to represent the surfaces implicitly using TSDF. For optimization, a batch of pixels are embedded in the form of rays according to the camera information. Adam optimizer is used to minimize the defined loss function with reference to the network parameter and camera information.

**Formulation of the network**

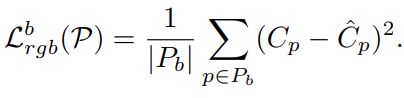
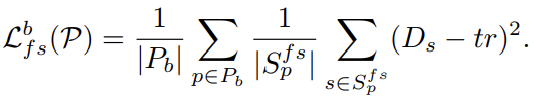
The way of querying the model is to encode the three-dimensional coordinate and direction using a sinusoidal position function. And the output of the network is a truncated signed distance value of the closest surface along with a radiance factor that represents the complete information of shape and colour. The input data of the network is bunch of rays that are generated from batches of pixels according to camera poses. Then the training process is achieved by optimizing the loss function explained in the next paragraph.

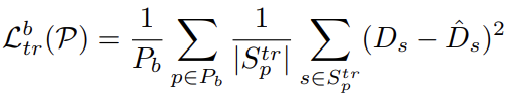
**Loss function**

The function is defined as



Where refers to the sum of squared difference of the predicted and ground-truth colour of pixels in each batch. measures the error related to the truncated distance of predicted surface and ground truth in free space. And measures the predicted error within truncated region. These loss functions are defined below:



In the above formulae, represents the number of rays used in each iteration, and denotes number of batches. For the demo, the parameters to generate the result are defined in the following table:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Parameter |  |  |  |  |  |  |  |
| Value | 0.1 | 10 | 6000 | 1024 | 5 | [-1,1] |  |

**Exporting meshes & Neural clean**

For validation and cross comparison between other methods, we extracted the mesh into a polygon file. The result mesh is generated by applying marching cube algorithm to the series of signed distance queries. However, the resulting model still contains some redundant parts that are out of the region of the reconstructed scene. In order to obtain a better result, we implemented a post-processing script that is attached to the rendering stage. The resulting method is called Neural clean and added to the validation stage.

Dataset

ScanNet is used for generating the final result in the demonstration. ScanNet contains scans of 707 labelled real-world indoor scenes with 2.5 million RGB-D images. Which is sufficient for both training and validating. Also, because the data are captured from real cases it contains flaws and noises that may happen in real life. It is useful for us to validate the performance and robustness because the robot would be used in real life.

The four scenes used for demo are scene50, scene84, scene580 and scene616. They covered a range of different scenarios and level of complexity. Performance of models will be tested on various of use cases for a better understanding of advantages and drawbacks of them.

The input data is a series of images and same amount of depth maps. In addition, camera intrinsic and pose information of each picture is also provided.

**Intrinsic 3D**

**texture map generation**

Recovery on texture and colour is a core novel feature of Intrinsic 3d, there are also lots of work done in the area of high-quality estimate on textures. Some works focus on generating consistent colour map directly from data source. For example, in a project led by Jeon, [1] they proposed a technique called texture coordinate that is designed especially for texture map. Klose et al. [2] also introduced filter to process the colour information in the spatial field. Besides, the research team led by Zhou [3] developed a way of overlaying the colour on the geometrical mesh precisely. Both camera information and two-dimensional deformation filed are used in their research. The focus of these methods is not only to align the RGB data to the meshes consistently with the input images. Colour-to-colour consistency is also an important requirement to texture recovery tasks.

Main Paper

<https://dazinovic.github.io/neural-rgbd-surface-reconstruction/static/pdf/neural_rgbd_surface_reconstruction.pdf>