

The experiments are performed on two normal inference tasks: normal inference based on depth image and guided normal inference based on RGB-D image. Prior works for normal estimation using very deep networks. For a single object surface normal detection as stated in this thesis, the given methods has a similar performance but only with 1/10 size.

The model is trained with PyTorch 1.10.0a0, CUDA 11.4.1, GPU with single NVIDIA GEFORCE RTX 3090.

1 GCNN model evaluation

In order to ease this issue, a light inpainting model has been trained based on the GCNN network with the identity architecture but the input and output, which takes the semi-dense light map as input and predict the fully dense light map as output.

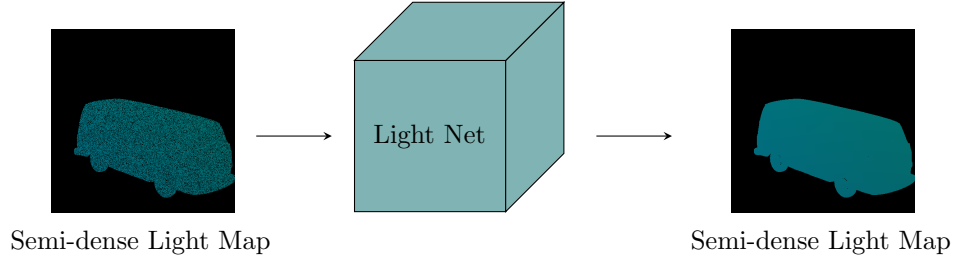


Figure 1: Light Net for light inpainting based on GCNN architecture.

2 Surface Normal Inference based on Depth Map

The goal of surface normal inference is to calculate the tangent surface normal map N from a single depth map D . A network named Gated Convolution Neural Network (GCNN) (??) is trained and is compared with similar approaches. We evaluate the performance of the model in dataset “synthetic50-5” introduced in ?? including 5K depth maps with size $512 \times 512 \times 3$ for training. All the depth maps add simulated noise as introduced in ?. The training pipeline use batch size 8, Adam optimizer (**adam**), learning rate of 1×10^{-3} , penalty-l2 loss. The output is directly the tangent normal in range $[-1, 1]$. The output and input has the same shape.

2.1 Qualitative Evaluation

The evaluation visualization on test dataset is shown in Figure 2. As shown in the picture, GCNN model Figure 3 zooms in the hindhead area of the dragon object which provides a closer comparison with the ground-truth.

The evaluation visualization on real dataset is shown in Figure 4

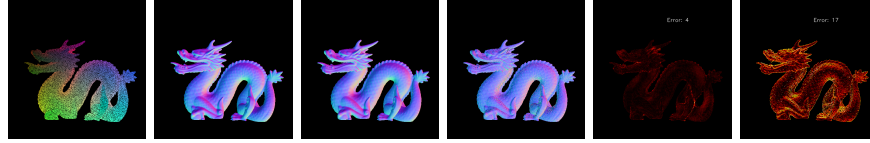


Figure 2: GCNN Normal Inference on Synthetic Dataset (object: dragon) From left to right: Input vertex map, ground-truth, GCNN normal map, SVD normal map, GCNN error map, SVD error map

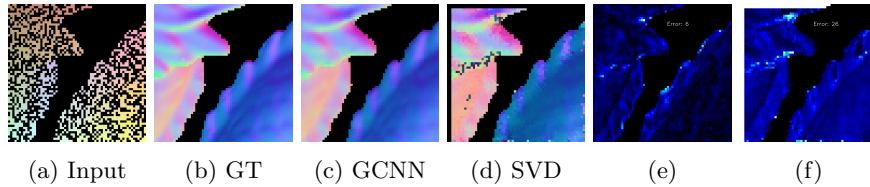


Figure 3: Zoom in of dragon object evaluation.

2.2 Quantitative Evaluation

The evaluation result on test scenes is shown in Figure 5. The GCNN based method has angle error between 5 to 15 degrees in both type of inputs. The error trends to higher with point number decrease. It is because the less points in the point cloud, the more detail is hidden due to the insufficient of resolution. Therefore the recorded surface based on the point cloud is more coarse, which also increase the difficulty of the normal inference.

2.3 Speed

3 Guided Gated Convolution Neural Network for Normal Inference

3.1 Light Net Evaluation

We evaluate the GCNN architecture based light net on the test dataset. The light net parameters are further used as the initial parameter of light branch of the Trignet.

A qualitative evaluation of light net is shown in Figure 6. The average angular error are lower than 0.3 degree on all of test cases. In this case, to distinguish the output and the ground truth is already not easily only use naked eyes.

Figure 7 uses a scatter plot to show the average angular error on 100 different test cases. The average pixel-wise angular error is 0.17 degree as shown in Table 1. An regression line has been added in the plot to analysis the tendency of the errors. The angular error slightly goes up when valid point number in the

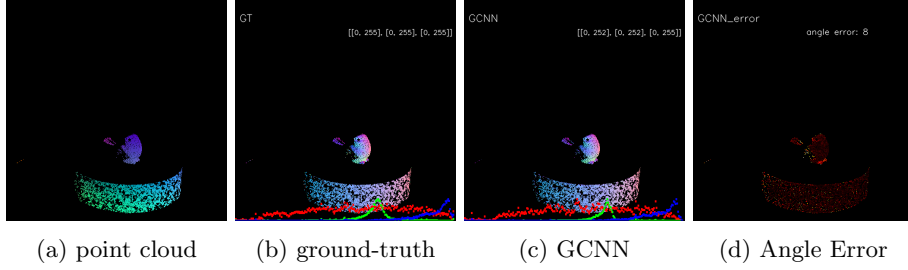


Figure 4: Evaluation on Real Dataset

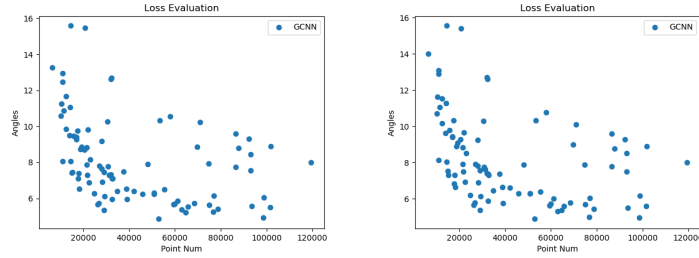


Figure 5: Evaluation of average angular loss on the whole test dataset with 90 scenes. The x-axis indicates the point number, the y-axis indicates the angles. The **Left** one using point cloud without noise, the **right** one has noise.

scene increase. It is make sense since the valid pixels are usually connected and concentrate in a single patch, the less of the area of the patch, which corresponding the number of the valid points, the less variation of the light direction among the pixels, thus better the evaluation.

3.2 Vertex-Image-Light Net (VIL Net) Evaluation

The qualitative evaluation of VIL net uses the model trained based on masked L2 loss.

3.2.1 title

3.3 comparison

From the Figure 11 we can observe the normal difference between ground-truth and GCNN predicted normals in another dimension. It separates the interval $[-1, 1]$, which is exactly the range of normal vector, to 256 sections. Then it counts the number of points locates in each section for 3 axes. The 3 axes are fitted quite well in most of interval but other than $[-0.25, 0.25]$ for x and y axes and interval close to -1 for z axis. Therefore a further constraint can be

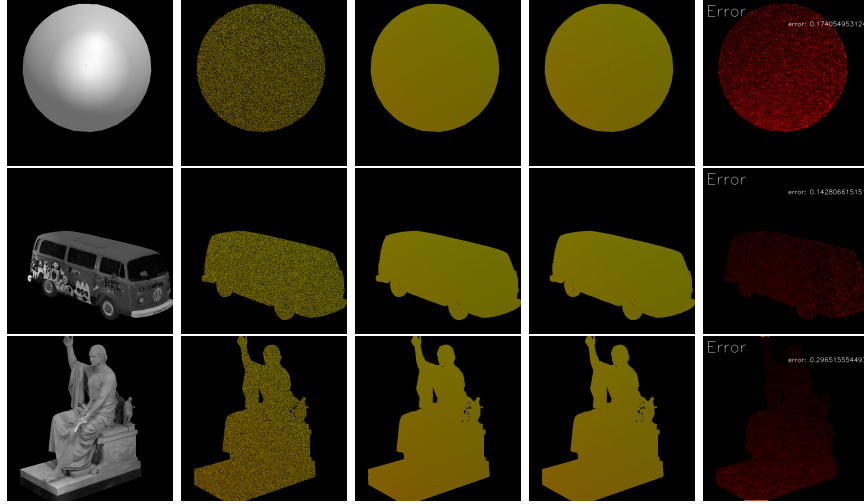


Figure 6: Qualitative evaluation of light net on Sphere, bus and Washington statue. From left to right, grayscale image, semi-dense light map(input), full-dense light map(output), full-dense light map(ground-truth), error map.

considered to the loss function related to the normal difference shown in this figure.

It is faulty that almost no normal has -1 z-component in GCNN predicted normal map. The reason?

Model	Angle	Time /ms	bz	lr-schedule	lr-df	l/i. Nr.	over-perform
LightNet	0.17	4.72					
GCNN	10.57	5.25	8	200,1600	0.5	0	-
an3	10.46	64.86	8	3,12,1000	0.5	1	yes
an3	10.81	65.34	8	10,1000	0.5	1	no(yes)
VIL-1	10.50	31.57	8	10,1000	0.5	1	yes
VIL-1	10.82	32.61	8	3,12,1000	0.5	1	no(yes)
VIL-3	10.79	54.76	8	100,1000	0.1	3	no(yes)
VIL-10	11.10	132.32	8	100,1000	0.1	10	no(yes)

Table 1: The error of models. The angle error is the average angle error of all valid pixels in the test case. The time unit is millisecond. bz is the batch size, lr-schedule is learning rate schedule. lr-df is learning rate decay factor, l/i. Nr is the number of light-image maps used for each scene

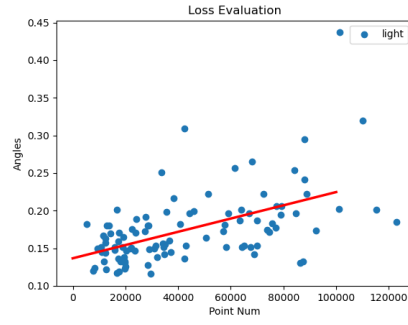


Figure 7: The evaluation on 100 test cases of Light Net.

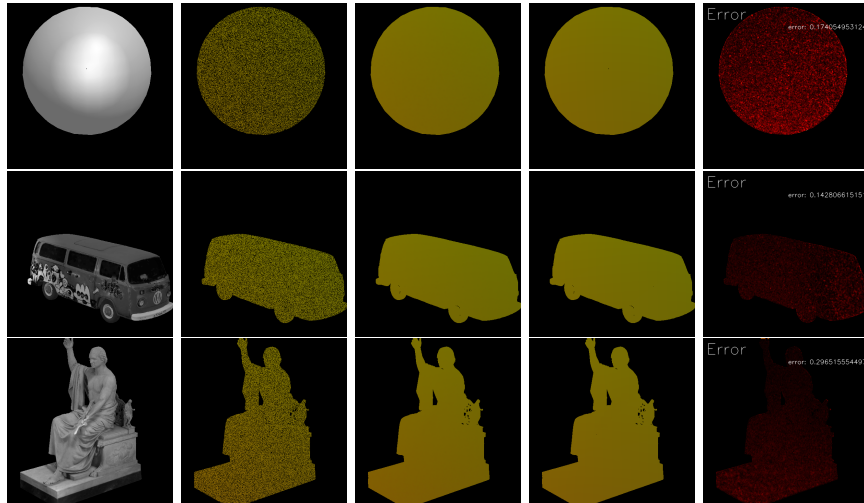


Figure 8: Qualitative evaluation of Trignet net on Sphere, bus and Washington statue. From left to right, semi 3D vertex map, full-dense normal map(output), full-dense normal map(ground-truth), error map.

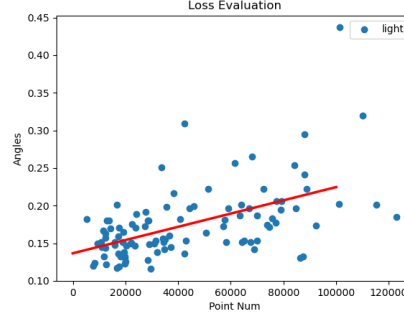


Figure 9: The evaluation on 100 test cases of Trignet.

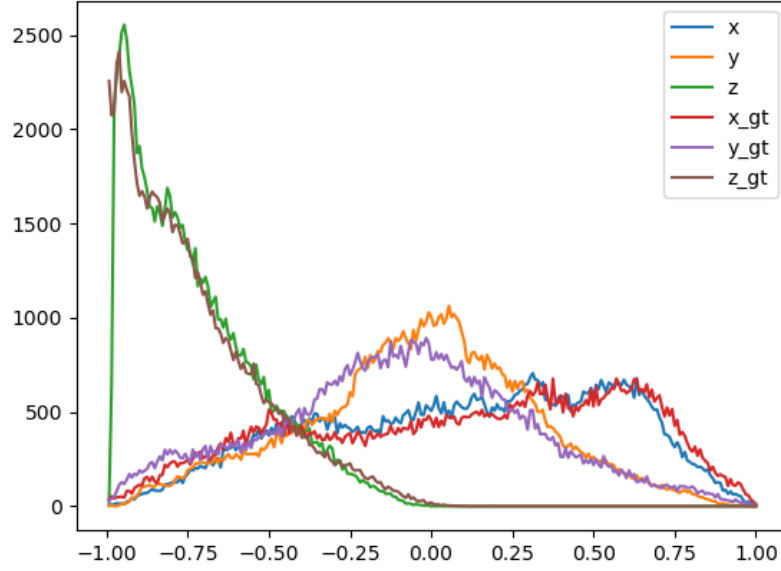


Figure 10: The normal difference of between GCNN and ground-truth in x, y, z-axis respectively. The y axis indicates the number of points, x axis indicates the value of normal in x/y/z axis. (The chart is based on the "dragon" scene showing above)

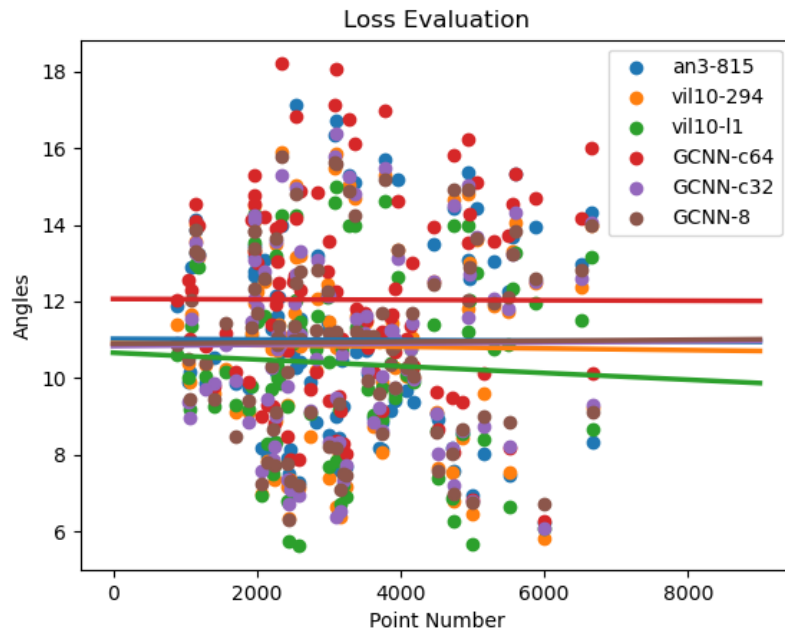


Figure 11: The normal difference of between GCNN and ground-truth in x, y, z-axis respectively. The y axis indicates the number of points, x axis indicates the value of normal in x/y/z axis. (The chart is based on the "dragon" scene showing above)