2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

A Supervised Approach to Predicting Noise in Depth Images

Chris Sweeney[1], Greg Izatt[1] and Russ Tedrake[1]

***Abstract*— Modern robotic systems are very complex and need to be tested in simulations with detailed sensor noise models to effectively verify robotic behavior. Depth imagery in particular comes with significant noise in the form of scene-dependent pixel-wise dropouts and distortions. Unfortunately, many depth camera simulations contain limited noise models, or can only support generating realistic depth images of simple scenes, which limits their usefulness in effectively testing perception algorithms. We propose a data driven approach to generate more realistic noise for complex simulated environments by using a convolutional neural network (CNN) to predict which pixels of a simulated noise-free depth image will not have returns (no-depth-return pixels, or NDP). We choose to focus on NDP here, as these dropouts are the most common and dramatic form of depth image noise. To train this network, we use reconstructed real-world scenes from the Label Fusion dataset to provide ground truth depth for each noisy depth image used to scan the scene. We use the resulting noise-free and noisy depth image pairs as labeled examples and train the network to predict which pixels of the noise-free image will be NDP. When used to post-process a simulation of a depth sensor, this system produces realistic depth images, even in cluttered scenes. To demonstrate that our approach successfully closes the reality gap for depth imagery, we show that the popular ICP algorithm for object pose estimation fails more realistically on our CNN-corrupted simulated depth images than on uncorrupted depth images and unsupervised domain adaptation baselines.**

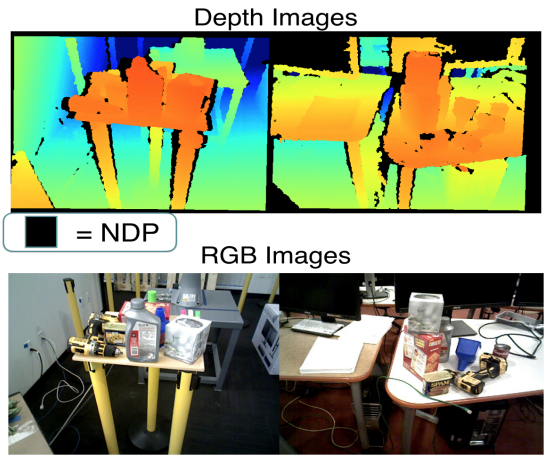


Fig. 1: Samples of registered RGB and depth images from the Label Fusion dataset. The black pixels in the depth images are NDP (No Depth Pixels). We model this type of noise and add it to depth image simulations.

I. Introduction

Simulated perception data allows robotic systems to be trained and tested efficiently and exhaustively. Days of labeled data can be simulated in a matter of seconds, allowing quick development and verification of robot behaviors. Unfortunately, today's simulations of robotic perception systems are too simplistic to guarantee any meaningful real world behavior. Robot simulation tools have struggled to close the reality gap that arises from the combination of complex environments with a diverse ecosystem of sensors and associated noise profiles [1]. Depth imaging is an area in robotic perception where the simulation reality gap is large, due primarily to the presence of complex camera, geometry, and material-dependent noise. Nonetheless, these sensors have become an important workhorse in robotic perception systems, as the high density of depth readings provide detailed scene geometry information that is highly valuable to manipulation and locomotion alike. Because of their particular popularity, we focus our discussion on depth cameras that operate via a *structured light* approach, though this approach is applicable to any dense depth camera that has NDP-like noise. Structured light cameras include the popular Microsoft Kinect and ASUS Xtion Pro. These cameras measure depth by projecting a series of infrared patterns onto a scene and using the perceived disparity between the patterns to reconstruct dense depth readings for each pixel in the resulting depth image. As a result of this imaging method, depth images from e.g. the Kinect show characteristic large-scale noise [2].

A given depth image can show both finite error in depth estimation and complete dropouts of certain pixels (No Depth Pixel, or NDP). We focus on only NDP in our analysis, as it is one of the most salient noise artifacts in depth imaging. There are at least two major sources of dropouts: interaction of the depth camera's projector-receiver pair with the geometry in the scene, and the interaction of scene illumination and material properties with the projected IR pattern. Complex object geometries including highly angled or discontinuous surfaces, from the perspective of the camera, result in chaotic noise artifacts that often result in NDP. Additionally, parallax from the distance between the infrared projector and infrared camera cause many near field objects to have NDP around their boundaries. As for object material properties, surfaces that are reflective, diffuse, and transparent further disrupt the projection and imaging process, resulting in both NDP and complex depth measurement errors. Real world scenes are cluttered and filled with these types of objects. Therefore, having more sophisticated simulations to mimic the noise in this regime is essential. Figure 1 shows depth images and their corresponding RGB images from the Label Fusion dataset, developed by Marion et al. [3], captured with the

[1]C. Sweeney, G.Izatt, and R. Tedrake are with the Computer Science and Artificial Intelligence Laboratory (CSAIL) at Massachusetts Institute of Technology, Cambridge, MA, USA {csweeney,gizatt,russt}@mit.edu

ASUS Xtion Pro. In cluttered scenes, like those found in the Label Fusion dataset, there are many noticeable NDP, particularly around objects that are close to the camera. This type of noise makes it difficult for robots to make sense of the world when measuring depth in a cluttered scene – for example, as we discuss in our pose estimation case study, NDP causes significant biases and errors when performing pose estimation of cluttered objects.

Most approaches to create realistic depth images rely on hand-created, limited noise models or constrain the problem to mapping real world depth image characteristics onto simulations for single objects, often in an unsupervised setting. However, using a supervised approach, we can model the most common noise artifact, NDP, for a more general set of real world scenes using a CNN. By leveraging a dataset that includes real depth images of a scene (obtained by 3D reconstruction) along with pixel-wise NDP labels, our network has the ability to predict NDP in very cluttered environments. Further, we focus on evaluating our method in a *task-sensitive* way by measuring the similarity of errors that the popular Iterative Closest Point (ICP) pose estimation algorithm makes on real data, and on simulated data corrupted by our CNN.

II. Related Work

Given the ubiquity of depth sensing in robotics, many researchers have investigated modeling and simulating depth sensors, with special focus on the popular Kinect camera. Structured light cameras like the Kinect contain noise caused by geometric and material artifacts in the scene being measured. Research into modeling depth camera noise can be categorized into *scene independent* and *scene dependent* models. The former relies on a general stochastic noise models applied to any scene being observed. The latter explicitly considers the scene being measured in the noise model. We review research for both types of models.

A. Scene Independent Models

Scene independent models provide easy-to-apply tools to recreate depth sensor noise. Researchers have investigated empirically deriving these models from studying Kinect noise in constrained environments. Khoshelham et al. [4] investigates how accuracy of the Kinect sensor's depth measurements degrades with distance. Choo et al. [5] and Nguyen et al. [6] both empirically measure lateral and axial noise distributions to improve Kinect modeling. While scene-independent models are efficient and easy to understand, they are, by nature, unable to capture the scene-dependent interactions that cause major spatially-correlated artifacts that plague most depth images.

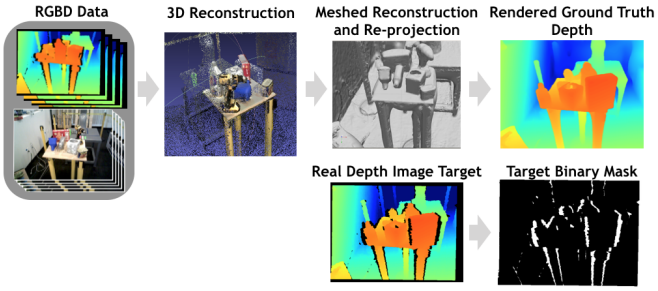
B. Scene Dependent Models

Scene dependent models directly use the simulated scene to generate realistic noise. Much of the needed information to predict depth camera noise is contained in the simulated scene. Frequently, scene geometry and composition interfere with the depth camera to cause NDP via high-angle surfaces or occlusions due to offset between the IR camera and projector. Additionally, highly diffuse, specular, and transparent surfaces disperse or absorb Kinect sensor infrared light, often resulting in no depth measurement returned [2]. Simulating the contribution of noise from all these external factors is difficult challenge, and can draw on both hand-crafted models and data driven analysis.

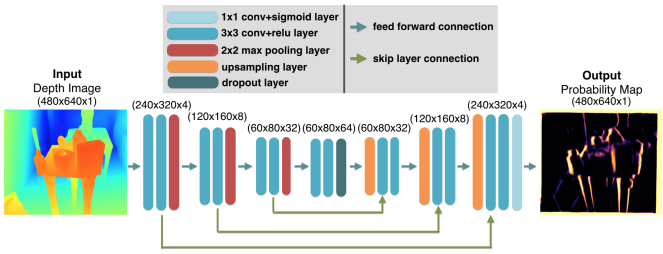
1) Model Based Approaches: With improved computer graphics engines, Landau et al. [7] and Planche et al. [8] have made efforts to more accurately capture depth camera noise by directly modeling the noisy intermediate infrared image of the structured light camera. These model based approaches simulate the method the depth camera uses to take measurements by rendering a noisy infrared image onto a simulated scene and processing it to create realistic depth images. Although model based approaches like these are powerful tools to organically simulate depth imagery, they depend heavily on the parameters used to model the structured light camera and its environment. Since structured light cameras like the Kinect have many unknown factors such as its infrared patterns and subtleties of the depth reconstruction algorithm, one often has to guess the characteristics of the underlying system or fine-tune these methods on real data. These model based approaches boast an incredible feature list and predict noise from complex properties of a scene, down to the level of lens distortion, specularity, and external illumination, but correspondingly come with increased requirements on detailed scene description before simulation is possible, and the growing problem of ensuring the simulated model remains true to reality.

2) Data Driven Approaches: Given an adequate training set, data driven approaches have the ability to find functions to approximate real world noise without being biased by assumptions made during modeling. The data driven approaches in this regime are often formulated as image to image translation and domain adaptation problems.

The task of generating realistic depth images from simulated images can be framed as a domain adaptation problem. Tobin et al. [9] uses extensive domain randomization to generate large amounts of non-realistic but extremely diverse simulated RGB images, and demonstrates that this non-realistic data is sufficient for some learning tasks. Shrivastava et al. [10] and Bousmalis et al. [11] on the other hand, train a model end to end to generate realistic images (RGB and Depth) from simulated images using Generative Adversarial Networks (GANs). More recently, the work of Zhu et al. [12] and Isola et al. [13] have gained significant popularity for their applicability to image to image translation problems in many different domains. [12] creates an impressive GAN (coined CycleGAN) that performs image to image translation without paired data, and [13] (coined Pix2Pix) similarly performs powerful image to image translation with paired data. Although these unsupervised methods using GANs are powerful and captivating, [10] and [11] are trained on adapting domains of *uncluttered* scenes with single objects. This is done to simplify the open-ended unsupervised learning problem. However, most indoor scenes contain many objects.



(a) Our completely automated process to create training data. We post-process the Label Fusion dataset to create all the necessary ground truth depth images for training. For any RGBD stream, we can reconstruct the scene into a point cloud. We then mesh the point cloud and render ground truth depth images from many viewpoints of a scene paired with the binary masked real depth image.



(b) Our CNN takes a simulated depth image and predicts a probability map of pixel-wise NDP probabilities. Our model contains about 100,000 parameters. We use a final 1x1 convolutional layer and sigmoid function to transform our feature maps into a pixel-wise probability image of NDP.

Fig. 2: Training data generation and model.

[12] and [13] are able to perform image-to-image translation for coloring or texture changes on a wide range of images, but tend to add unwanted distortions when corrupting images as needed to add noise to a depth image. Since we have domain specific knowledge that for depth camera noise, NDP is the major artifact, we can use a much more targeted and lightweight approach without introducing any unwanted distortions into a simulated depth image. Therefore, we solve a supervised learning problem, employing a CNN to predict NDP noise for complex simulated scenes.

III. Technical Approach

We present a novel technique to predict noise in simulated depth images using a CNN architecture. Our focus is directed at predicting NDP as a supervised pixel-wise binary classification task. Using real depth images as labels, we train on reprojected depth from the 3D reconstructed mesh of a scene. In order to model NDP from a wide set of noise artifacts, a diverse set of training data is required. Our method can quickly and automatically generate training data from any RGBD dataset.

A. Data Generation

In order to capture the necessary training data for our supervised learning problem, we set up a graphics pipeline that takes as input an RGBD stream and outputs reprojected depth images from the scene's reconstructed mesh. We train on depth images as they contain useful geometric information for determining whether or not a given pixel is NDP. RGB images can also contain information about the geometry of the scene. However, RGB images also contain a lot of superfluous information to the NDP prediction from scene geometries task such as color and lighting. To avoid overfitting to the non-relevant data, we constrain our model to process only rendered ground truth depth images.

Creating ground truth reprojected depth images require post-processing RGBD streams from the Label Fusion dataset. First, we create a 3D reconstructed point cloud of the scene from the Label Fusion RGBD stream using Elastic Fusion, developed by Whelan et al. [14]. Next, we turn the reconstructed point cloud of a scene into a mesh representation using Poisson Surface Reconstruction developed by Kazhdan et al. [15]. Then for each RGB and depth frame of a particular scene, we simulate the corresponding viewpoint of the reconstructed mesh. From here, we render the current view of the mesh into a depth image. Each rendered depth image is paired with its real depth image, binary masked to form NDP labels. Figure 2a describes our pipeline for generating this training data. Given the RGBD data of a scene, our pipeline is completely automated. There are many large RGBD databases that already exist (Silberman et al. [16] and Glocker et al. [17] to name a couple) and can be used with no human in the loop to create the necessary training data.

B. Our Model

NDP in a cluttered scene is very complex, and not effectively modeled by traditional stochastic models. We use a CNN network architecture inspired by Ronneberger et al. [18] to carry out the complex NDP prediction task. We model the NDP prediction for complex scene geometries as a pixel-wise binary classification problem using a CNN. Given a ground truth depth image, our network can learn the complicated features relevant for NDP prediction. Our labels for each ground truth depth image are the real world depth images where all the NDP are mapped to a 1 and every other pixel mapped to a 0.

Given the input rendered depth images, the network encodes the information in these images into a high level representation for predicting NDP. It then decodes this representation through up-sampling these high level representations and merging with higher resolution features of the images using skip layer connections. Figure 2b shows a visual depiction of our architecture. Our model outputs a probability map where each pixel in the image describes the probability of a NDP. We use this probability map to instantiate a mask to apply to a simulated depth image.

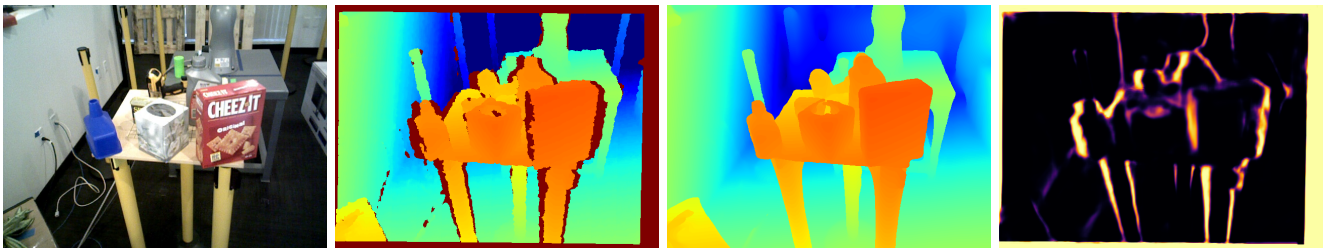


Fig. 3: *From left to right*: RGB image, real depth image, rendered depth image from the reconstructed scene mesh, CNN predicted NDP probability map. The depth images are color coded as a heatmap with red being near distances and blue being far distances.

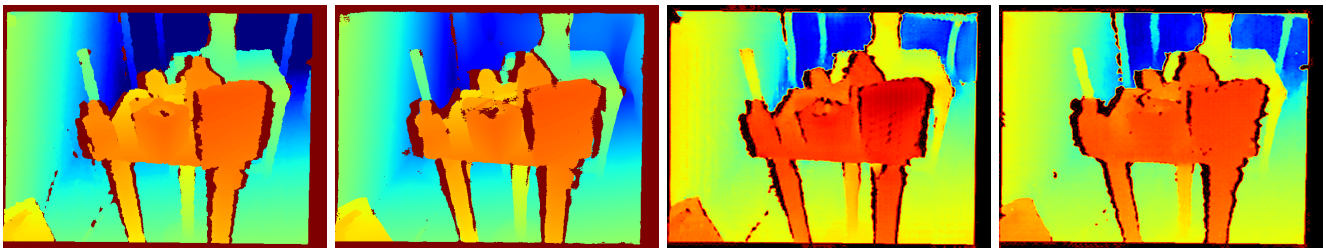


Fig. 4: *From left to right*: Real depth image, depth image with NDP predictions added to the simulated depth image from sampling the probability heatmap, depth images produced by CycleGAN and Pix2Pix. Both CycleGAN and Pix2Pix were given the reprojected depth image as input. Our method captures much of the real world depth image noise without adding unrealistic distortions seen in the unsupervised baselines. The unsupervised baselines distort things like depth scaling resulting in a more red depth image.

C. From NDP Probabilities to Simulated Depth Image

We use the NDP probability map from our CNN to form a image mask to corrupt a simulated depth image. Since NDP noise has a high frequency, scene independent component (often perceived as the characteristic *flickering* seen in depth imaging), we sample the probability map using a correlated noise process. For pixels in the NDP probability map with probability higher than some threshold (0.5 works well in practice) we mark the corresponding pixel in the image mask. For pixels in the NDP probability map lower than the threshold, we sample the probabilities using a Perlin correlated noise process, and mark the corresponding pixels in the image mask. Finally, we corrupt the simulated depth image by applying this mask, setting all pixels corresponding to marked pixels to 0 or NDP.

IV. Results

A. Qualitative Experiments

We evaluate the realism in our simulated depth images by visually comparing them to real depth images and domain adaptation baselines. Figure 3 shows an RGBD pair, the corresponding reconstructed depth image, and NDP predicted probability map. Figure 4 shows side by side comparisons of the real depth image, simulated depth image by sampling the NDP probability map, a depth image created by CycleGAN [12], and a depth image created by Pix2Pix [13]. Both the CycleGAN and Pix2Pix model were trained on the same Label Fusion training set as our model. From a visual perspective, our model seems to be able to best capture the complex NDP patterns seen in real depth images without compromising other parts of the depth image with unwanted distortions as seen in the images produced using CycleGAN and Pix2Pix.

B. Quantitative Experiments

We evaluate our method quantitatively in two ways. First we evaluate the ability of our model to predict NDP from the classification perspective and expand on its ability to generalize to novel scenes. Then we evaluate how well our simulation prepares robotic tasks for the real world by observing relative pose error statistics after applying the pose alignment algorithm, Iterative Closest Point (ICP). We compare relative error metrics of object poses in our NDP predicted depth image's projected point cloud to the poses of the same objects in the real depth image's point cloud after ICP alignment and baseline our approach on a simple NDP prediction scheme, the CycleGAN baseline, the Pix2Pix baseline, and perfect uncorrupted simulation.

1) NDP classification: Our NDP prediction CNN is lightweight, containing only about 100,000 parameters. The average feed forward inference time for NDP prediction is 13ms on a GeForce GTX 980 GPU. Thus, our model is well below the frame rate of a typical depth camera (30ms) and can be put at the end of a real time simulation pipeline with no large performance penalty.

To test the accuracy of our model, we use a 80%-20% train/test split of the Label Fusion dataset and plot the resulting ROC curves for NDP binary classification for four different models on the test set. Our curves are shown in Figure 5. We compare our model against three other baselines: a simple NDP prediction scheme that predicts NDP for pixels with normals that are too perpendicular to the view direction; CycleGAN; and Pix2Pix. The CycleGAN model was trained on the Label Fusion training set ignoring our pixel-wise image labeled pairs, and the Pix2Pix model used the paired images in our training set. Our method clearly outperforms the simple NDP prediction baseline in

terms of its sensitivity-specificity trade-off. Since CycleGAN and Pix2Pix produce deterministic results, their NDP binary classification performances exist as points in Figure 5. For a constant false positive rate of 10%, our method achieves a true positive rate of 85% while CycleGAN achieves a true positive rate of 67%. With a false positive rate of 4%, our method achieves a true positive rate of 80% while the Pix2Pix baseline achieves a true positive rate of 71%. With respect to NDP classification, our method outperforms the unsupervised and simple model based alternatives and also has the flexibility to change classification thresholds to trade-off true and false positive rates, unlike the baseline unsupervised methods.

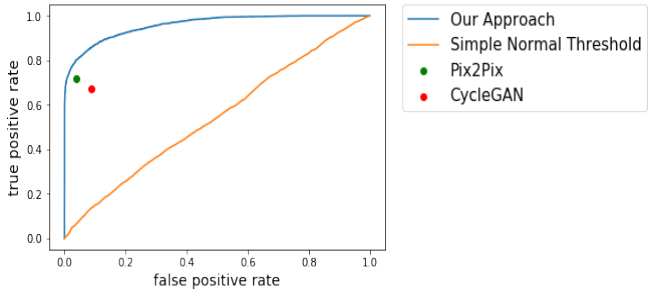


Fig. 5: ROC curves showing the NDP binary classification performance of our approach and three baselines. Our method achieves the best performance on predicting NDP, without falsely classifying too many pixels. The unsupervised baselines are points on the ROC curve as they produce deterministic images.

Our model performs well for the types of scenes often seen in robotic manipulation tasks. These types of scenes tend to be cluttered environments with many near-field objects. The Label Fusion dataset contains over 150 scenes with subsets of 12 known object in various orientations and positions. Therefore our performance on the Label Fusion dataset suggest, at a high level, we can achieve good performance for scenes in this regime.

In our evaluation of RGBD camera noise, we found that noise models change considerably between different depth cameras. (As a concrete example, parallax errors scale with projector-receiver baseline, which differs between camera model.) Fortunately, to generate a reasonably good NDP prediction model for a new depth camera, only small amounts of data are needed. We investigate the generalizability of our method in data limited environments to show that the training data for our method could be easily and quickly collected for a new depth camera.

We repeat our NDP classification analysis for a model trained on images collected from one scene (approx. 4000 images) of the Label Fusion dataset and show one can get a pretty good degree of generalization with very limited training data. Results are plotted in Figure 6.

After training the NDP model on a single scene log (log0 in Figure 6), one can apply the model to environments where similar near field objects are put in many different configurations for a small penalty in performance. The limited amount of data needed suggests that training data could be collected as part of a calibration stack for robotic manipulation.

2) Pose Estimation Case Study: To evaluate whether our simulation captures the relevant noise to close the reality gap, it is important to evaluate how well our simulation prepares a robot for real world tasks. To quantify our simulation's task-relevant similarity to the real world, we inspect the relative error statistics of a common object pose estimation algorithm when applied to simulated and real depth images. Our algorithm of choice – Iterative Closest Point (ICP) [19] – is a local optimization method for object pose estimation in point clouds. This method serves as a good indicator for our simulation's ability to truly close the reality gap, as ICP is applied extremely broadly in robotic perception – for example, for object localization in robot manipulation (as in [20] and [21]), and for scan alignment for localization and mapping (as in [22]). We report relative percentage absolute difference between the real depth image simulation and our simulation and baselines for mean magnitudes and variances of the pose estimation error.

We now explain our evaluation pipeline. For a given simulated scene created from Label Fusion, we compare (1) an uncorrupted simulated depth image; (2) our NDP predicted depth image; (3) a NDP predicted depth image using the simple NDP prediction scheme; (4) the CycleGAN predicted depth image; (5) the Pix2Pix predicted depth image; and (6) the corresponding real world depth image. We then create six point clouds by projecting each of these depth images into 3D space. Next, we render point clouds of every object in the scene (for which we have the Label Fusion ground truth poses), slightly perturb their poses and then use ICP to realign the objects with each of the various point clouds. We calculate the mean and variance of Euclidean and rotational distances between the post ICP poses of the objects and their ground truth poses in each simulated point cloud. The Euclidean and rotational error metrics are defined below.

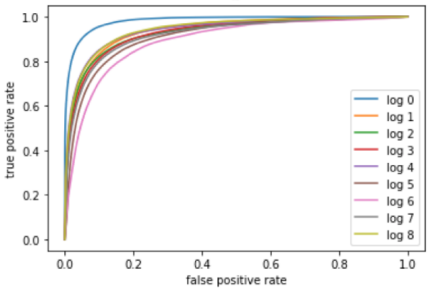
$$d(q_1, q_2) = 1 - \langle q_1, q_2 \rangle \tag{1}$$

$$d(p_1, p_2) = \sqrt{(p_{1_x} - p_{2_x})^2 + (p_{1_y} - p_{2_y})^2 + (p_{1_z} - p_{2_z})^2} \tag{2}$$

Eq. 1 is the rotation distance metric and Eq. 2 is the position distance metric. q_1 is the post ICP object quaternion and q_2 is the ground truth quaternion. p_1 and p_2 follow the same scheme but are the 3D coordinates of an object.

We show that the post ICP error statistics of objects aligned in our NDP predicted depth images are much closer to the real world post ICP error statistics than the errors for the pure simulated depth, the simple NDP prediction scheme and the unsupervised baselines.

We fit objects to point clouds from 15 various Label Fusion scenes each containing 5 different objects and report the mean and variance of distance errors relative to the errors of the real depth image in Table 1. The CycleGAN and Pix2Pix baselines have more drastic errors than the other experiments due to unrealistic depth distortions learned during training. Pix2Pix leverages paired image data like our approach, but has unrealistic errors with respect to its



Dataset	Area under ROC
log0	0.982
log1	0.939
log2	0.928
log3	0.930
log4	0.940
log5	0.912
log6	0.891
log7	0.921
log8	0.943

Fig. 6: *Left*: Sample images of 9 similar scene logs from the Label Fusion dataset. The upper left image (log0) is a sample from the scene used to train the data limited model. *Middle*: ROC curves for NDP classification performance on 9 similar Label Fusion scenes (labeled log0-log8). The model was trained on only 4000 images from data log 0 (blue curve), and is able to generalize well to the other scenes. *Right*: Area under ROC curve for the 9 scenes.

Methods	**Euclidean Distance Mean**	**Euclidean Distance Variance**	**Rotational Distance Mean**	**Rotational Distance Variance**
Perfect Sim	3.90%	18.4%	77.8%	49.8%
Simple NDP	3.60%	18.0%	77.2%	49.7%
Cycle GAN	62.3%	196%	227%	95.8%
Pix2Pix	187%	179%	53.8%	99.3%
Our Method	**3.60%**	**6.90%**	**3.74%**	**0.10%**

TABLE I: Table showing relative % error between various depth simulation schemes and the real world for various ICP error statistics. In terms of mean Euclidean and rotational ICP error, our method has the most realistic performance, having small differences with real world ICP errors. Other unsupervised baselines like CycleGAN and Pix2Pix distort the depth images resulting in unrealistically large errors with real world ICP errors.

Euclidean and rotation post ICP distances. CycleGAN does not used paired data, so it is not surprising that it contains more unrealistic distortions causing very large and chaotic rotational errors. While CycleGAN and Pix2Pix have greater representational power than our more handcrafted method, their unsupervised nature makes it harder to trust them as a source of realistic depth imagery in robotics.

Since objects in our simulation have similar post ICP error statistics with the real world for a fundamental robotic object pose estimation algorithm, we are confident that more complex robotic tasks verified in our simulation will be more likely to succeed in the real world. We believe that our analysis using ICP error signals is a useful tool for the evaluation of robot simulations with a focus on downstream tasks in manipulation and locomotion.

V. Discussion

Our method relies on meshing 3D reconstructions of a scenes and rendering depth images from them. However, meshing and 3D reconstruction performance decreases with increasingly cluttered scenes, causing objects in the scene to be warped. Since rendered images of these suboptimal meshes serve as ground truth images for our supervised learning task, our model is slightly biased to perform better on these 3D reconstructed meshes rather than real simulated environments. However, RGBD data is easy to create and use, opening up the possibility of quickly collecting large training sets for our method. While the geometries of objects in a scene contribute significantly to NDP, the materials properties of those objects also cause prevalent noise.

For the supervised regime, to our knowledge, there do not exist any large datasets that have labeled material properties for a given scene. The most common tools we have to measure object material properties involve the Bi-Directional Reflective Function. Using BRDF-like scanners on every object in a scene is impractical and time consuming for building large annotated datasets. However, recent approaches like Park et al. [23] show that lighting properties of objects can be captured accurately and efficiently using commodity IR depth sensors. For an exhaustive modeling of depth camera noise, it is essential to infer and parameterize the material properties of objects in a scene in a concrete way, but efficient way.

While this work exclusively focuses on NDP, there are other types of prevalent noise in depth imagery. For example, when measuring depth from transparent objects, structured light depth cameras often return blotches of pixels with incorrect depth values. Unlike NDP, robots have no clear way of making sense of this noise. For future iterations of depth camera simulations, it is important to capture other types of depth errors to make robots more robust to interacting with many different objects.

VI. Conclusion

We have introduced a new data driven approach for creating realistic depth images using a convolutional neural network. Training our CNN on depth images rendered from the 3D reconstructed mesh of a scene, we are able to accurately predict NDP in cluttered and complex environments. Finally, we showed that adding our NDP predictions to a depth camera simulation decreases the reality gap when performing an ICP object pose estimation task, demonstrating that our method advances our goal of providing more realistic and trustworthy sensor simulation.

VII. Acknowledgment

This work was supported by NASA - Johnson Space Center; Award No. NNX16AC49A, and a National Science Foundation Graduate Research Fellowship under Grant No. 1122374. The views expressed in this paper are those of the authors themselves and are not endorsed by the funding agency.

REFERENCES

[1] K. Bousmalis, "Closing the simulation-to-reality gap for deep robotic learning," Oct 2017. [Online]. Available: https://ai.googleblog.com/2017/10/closing-simulation-to-reality-gap-for.html

[2] T. Mallick, P. P. Das, and A. K. Majumdar, "Characterizations of noise in kinect depth images: A review," *IEEE Sensors journal*, vol. 14, no. 6, pp. 1731–1740, 2014.

[3] P. Marion, P. R. Florence, L. Manuelli, and R. Tedrake, "Label fusion: A pipeline for generating ground truth labels for real rgbd data of cluttered scenes," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8, 2018.

[4] K. Khoshelham and E. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," in *Sensors 2012, 12, 14371454. 2013*, p. 8238.

[5] B. Choo, M. D. DeVore, and P. A. Beling, "Statistical models of horizontal and vertical stochastic noise for the microsoft kinect," in *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, Oct 2014, pp. 2624–2630.

[6] C. V. Nguyen, S. Izadi, and D. Lovell, "Modeling kinect sensor noise for improved 3d reconstruction and tracking," in *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*, Oct 2012, pp. 524–530.

[7] M. J. Landau, B. Y. Choo, and P. A. Beling, "Simulating kinect infrared and depth images," *IEEE Transactions on Cybernetics*, vol. 46, no. 12, pp. 3018–3031, Dec 2016.

[8] B. Planche, Z. Wu, K. Ma, S. Sun, S. Kluckner, O. Lehmann, T. Chen, A. Hutter, S. Zakharov, H. Kosch, *et al.*, "Depthsynth: Real-time realistic synthetic data generation from cad models for 2.5 d recognition," in *3D Vision (3DV), 2017 International Conference on*. IEEE, 2017, pp. 1–10.

[9] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 23–30.

[10] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2107–2116.

[11] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 2, 2017, p. 7.

[12] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 2242–2251.

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2017.

[14] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "Elasticfusion: Real-time dense slam and light source estimation," *The International Journal of Robotics Research*, vol. 35, no. 14, pp. 1697–1716, 2016.

[15] M. Kazhdan and H. Hoppe, "Screened poisson surface reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 32, no. 3, p. 29, 2013.

[16] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision*. Springer, 2012, pp. 746–760.

[17] B. Glocker, S. Izadi, J. Shotton, and A. Criminisi, "Real-time rgb-d camera relocalization," in *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, Oct 2013, pp. 173–179.

[18] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[19] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor Fusion IV: Control Paradigms and Data Structures*, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586–607.

[20] T. Schmidt, R. A. Newcombe, and D. Fox, "Dart: Dense articulated real-time tracking." in *Robotics: Science and Systems*, 2014.

[21] J. M. Wong, V. Kee, T. Le, S. Wagner, G.-L. Mariottini, A. Schneider, L. Hamilton, R. Chipalkatty, M. Hebert, D. M. Johnson, *et al.*, "Segicp: Integrated deep semantic segmentation and pose estimation," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 5784–5789.

[22] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 2011, pp. 127–136.

[23] J. J. Park, R. Newcombe, and S. Seitz, "Surface light field fusion," in *2018 International Conference on 3D Vision (3DV)*. IEEE, 2018, pp. 12–21.