Fast Convolutional Neural Network for Depth Image Inpainting

Rajer Sindhu, Vihaan Mattu, Reyansh Sachdeva

1 abstract

In this paper we propose a dynamic method for depth imaging recovery based on as mixture of signals that are integrated into a novel convolutional network. The proposed convolutional network contains three important layers, which are used to project high dimensional data into a much lower dimension of space and can be flexibly applied for missing data completion and reconstructing a complete depth image with consistent distribution. In our framework, the input include depth image and color image; by using CNN learning, the framework output the corresponding recovered depth image. To filter out unnecessary details and improve the edges of the grey image, we designed a preprocessing procedure with data augmentation strategy to regenerate and increase the essential training data. Also, we propose a adapted weight loss function for the neural network training to improve the learning efficiency. This algorithm has been tested on public dataset and shows satisfactory results on many challenging scenarios. We compare the output image with the ground truth and demonstrate the accuracy can reach over 90 percent on the estimated depth map.

2 Introduction

Depth cameras have become more portable and affordable [? ?] giving us an opportunity to use them to improve performance in vision associated matters such as recognition, tracking, segmentation, and reconstruction. To solve issues like black holes around edges and noise being much stronger than the color image, depth image denoising and enhancement are applied. Here, the denoising step is used to fix corrupted isolated pixels and small regions. The enhancement step aims to improve image details, especially the edges of the depth image [?]. Multiple image processing methods such as joint bilateral filtering [???], image reconstructing [?], spatial temporal relationship [?], cost-volume [?], wavelet tight frame [?] and low rank matrix [? ?] were developed and they all based on color-depth relationship. Deep learning [? ? ? ?]. The improvement of the image quality is shown in Fig-1. In order to improve the quality of the depth image, denoise and enhance convolutional

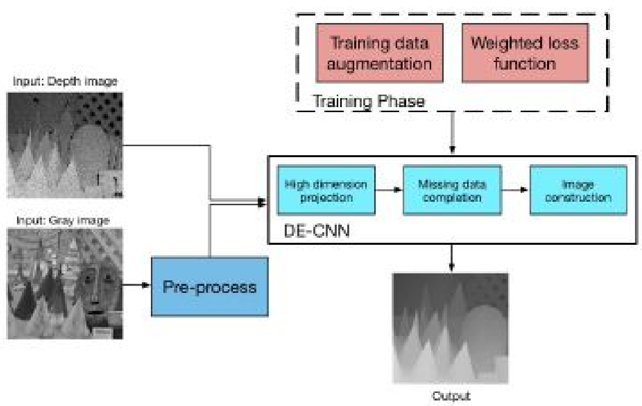


Figure 1: The flowchart of DE-CNN based depth image denoising and improvement

neural network (DE-CNN) was proposed. DE-CNN is a pixel-wise generative network with three layers where each layer has a different purpose. The first layer is for high dimensional projection, second is for missing data completion and third is for image reconstruction [?].

3 DEPTH DENOISE AND ENHANCE CONVOLUTION NEURAL NETWORK (DE-CNN)

Pixel-wise generative model is used to solve the denoising and enhancement problem depth images. By using SRCNN [?] and FCNN [?], the convolutional neural network and design the DE-CNN framework were produced. By using DE-CNN three layer, we do the following: 1) The full connection layer is not employed [?]. 2) Add the max pooling to screen out to certain corrupted values to denoising [?]. 3) Depth color image relationship and emphasis on the edges [??]

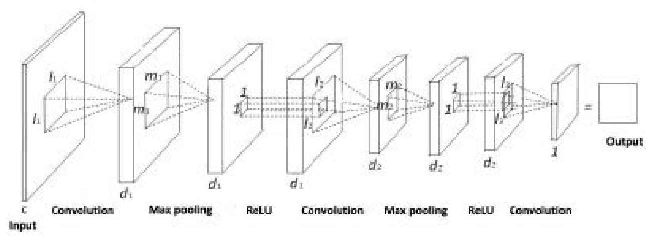


Fig. 2. The structure of DE-CNN

Figure 2: The structure of DE-CNN

DE-CNN framework The three layers of DE-CNN are described below: High dimensional projection: This layer set consists of a convolution layer, a max pooling layer and a rectified linear unit (ReLU) layer where the convolutional layer can extract invisible existing information, while the max pooling process helps to screen out those black holes and noisy parts [?]. Missing data completion: This layer also consists of a convolutional layer, a max pooling layer and a ReLU layer. The convolutional layer fills up the missing depth image, and the max pooling layer discards the visual information and keeps the depth data [?] Image reconstruction: In this layer, the convolution step is only used to generate the final output of the depth image because all the other layers (max pooling and reLU) can waste this layer.

Loss function definition The loss function is used to focus on edges and it treats every part equally. The function is defined as the following where M is the weighted map and IO and IG represent the network output and ground-truth images individually [?]:

$$f_loss = ||M.(I_O - I_G)||^2$$

Next we set the values around the edge close to 1 and the values in the smooth parts can be lower after getting the edge information from the ground depth image.

4 DATA PREPROCESSING AND AUGMENTATION

The relationship between color and depth are used in [?]. To complete the hole-filling and denoising tasks in a depth image, we use depth and gray images collectively.

Gray image pre-process To eliminate useless parts and highlight important details, gray image pre-processing method is used. In Figure 3. We can see the six steps of pre-processing: intensity equalization, bilateral filtering, edge extraction, watershed segmentation, segment average padding and intensity quantization [?]. Out of these six steps, watershed segmentation and segmentation average padding are used to grab the unique intensity pixels into the same region then all the useless details are faded and the edges are highlighted.

Depth image pre-processing and training data augmentation The first step is to drop training patches whose corresponding groundtruth data contains black areas ?. Figure-3 shows the pre processing procedure. Another issue is the having limited number of samples majorly black hole

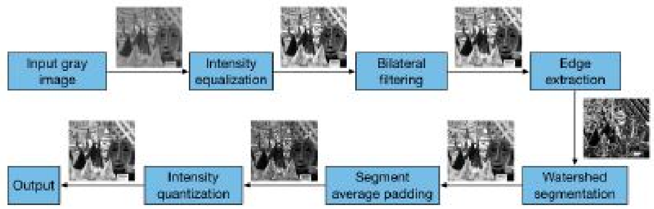
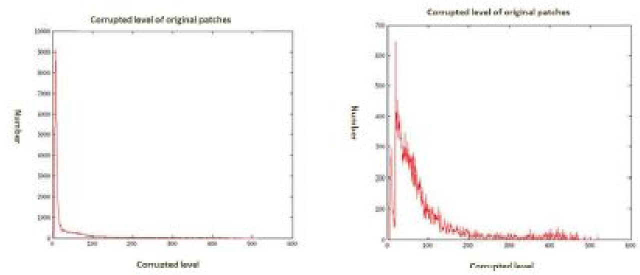


Figure 3: The pro-process procedure of gray images.

samples. Filling and padding of the black area is going to be the most challenging part of the depth image enhancement. Figure 4 shows analytics of all the black pixels connected in every patch. There are too many patches with 15 connected black pixels and in order to increase the patches ?, we use a hyperbolic curve for probability distribution to identify the set.. ?1 and ?2 (0 ¡ ?1 ¡ ?2 ¡ 1), patches with pi less than ?1 are eliminated by this probability; patches of larger than ?1 but less than ?2 are kept by this corresponding probability; patches with greater than ?2 are duplicated according to their pi ?. We will now propose a new strategy by duplicating individual patches by rotating specific chosen patch by 90 degrees, 1800 degrees or 270 degrees. The result is shown in Figure 4b.



(a) (b)

Figure 4: The histogram of the number of connected pixels in the training data (a) and processed data set (b).

5 Experiments

We will now detail the structure of DE-CNN and compare our methods with two state-of-the-art depth denoising and enhancement methods in terms of speed, PSNR, and visual effects [?]. The Middlebury dataset [?] [?] consists of 30 pairs of ground-truth depth and color images [?]. In order to simulate the noisy pictures, the authors manually added black holes in the depth images.

DE-CNN framework evaluation In this experiment, we will use two images as test images and 28 of the Middlebury set as the training data. We position the first unit as a convolutional layer of size 1 9 9 128, a 5 5 max pooling layer and a ReLU layer. The second unit consists of a 1281164 convolutional layer, a 3 3 max pooling layer and a ReLU. A single 64551 convolutional layer acts as the last unit [?]. Well use two aspects two aspects: single depth input vs. joint depth-RGB input; Euclidean loss function vs. edge based weighted loss function to compare the structure.

Input data comparison We will now compare the following inputs: single depth, joint depth and color input. The large black hole areas have been better padded with clear edges, such as the long brush in test figure one [?]. Figure-5 and Table-1 shows the input validation.

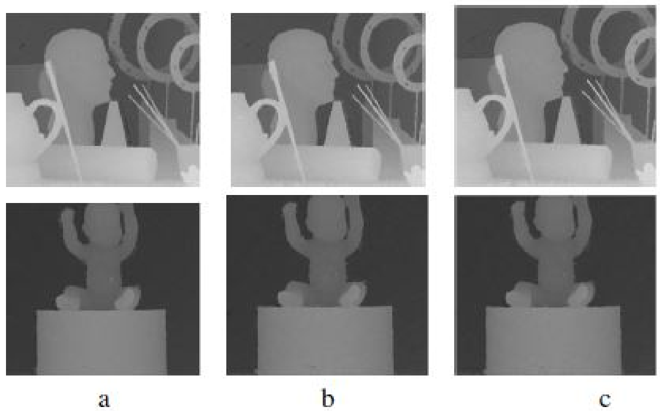


Figure 5: The DECNN setting comparison: (a) single depth input; (b) joint depth and RGB input; (c)joint input with preprocessing and weighted loss function.

PSNR (dB)	Depth Input	Joint Inputs	Weighted Loss Function
Test One	32.56	33.46	33.68
Test Two	38.96	39.02	39.18

Figure 6: The PSNR comparison of different settings on two testing figures.

Loss function To focus on the black holes and edges, we will use edge based weight maps as weighted loss function. In summary, these experiments have demonstrated the effectiveness of our network design and data preparation [?].

Comparison with other algorithms Now we will compare DE-CNN with another two recent algorithms that deliver the best results among others. We denote the low rank matrix completion method [?] as LRMC and the data-driven tight framework [?] as DDTF in the following [?]. To compare, well use 30 images in the Middlebury dataset and compare the methods for their quality, PSNR and efficiency. Quality: The results are shown in Figure 6 an the quality is almost same except

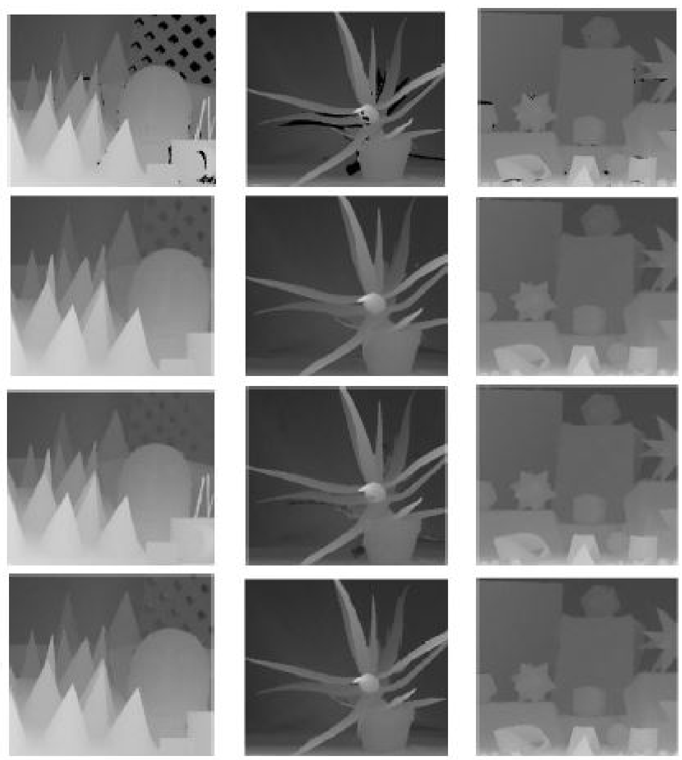


Figure 7: The visual quality comparison of our DE-CNN with two other methods. First row: input depth image; second row: LRMCs results; third row: DDTFs results and fourth row: proposed DE-CNN results.

the edges where the DE-CNN are sharper than the other methods. PSNR: All the 30 images PSNR results are summarized in Table 2. Speed: In order to process the a 352395 depth image using the NVIDIA TITAN X GPU, DE-CNN took 0.033 seconds.

Middlebury dataset	Flower			Sculpture			Infant 1		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	**36.24**	35.96	35.63	**33.84**	33.20	32.29	**40.28**	38.85	38.92
Middlebury dataset	Infant 2			Infant 3			Book		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	41.92	42.59	**42.99**	39.76	42.32	**43.33**	41.37	40.75	**42.12**
Middlebury dataset	Bowling 1			Bowling 2			Cloth 1		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	37.52	**38.22**	38.00	38.71	**39.37**	38.18	45.01	46.24	**46.86**
Middlebury dataset	Cloth 2			Cloth 3			Cloth 4		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	41.45	**42.35**	42.33	**42.75**	42.37	42.16	**39.16**	37.43	37.57
Middlebury dataset	Cone			Toy 1			Clay pot		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	39.16	**39.85**	38.56	40.67	41.91	**42.13**	37.62	42.73	**43.77**
Middlebury dataset	Toy brick 1			Toy brick 2			Window		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	**39.96**	38.79	39.37	**39.91**	38.56	39.32	38.11	**38.49**	37.91
Middlebury dataset	Bag 1			Bag 2			Origami		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	**40.39**	39.82	39.06	**39.41**	38.57	38.52	41.07	**41.95**	41.67
Middlebury dataset	Board game			Folder			Elk		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	39.32	39.41	**40.02**	42.72	42.04	**43.34**	35.36	35.46	**35.90**
Middlebury dataset	Stone 1			Stone 2			Toy 2		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	43.39	45.26	**45.49**	43.27	46.36	**46.58**	39.77	40.74	**40.80**
Middlebury dataset	Wood			Board			Newspaper		
Method	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF	DE-CNN	LRMC	DDTF
PSNR	**40.84**	39.28	40.80	40.46	39.85	**41.04**	**41.36**	41.32	41.18

Figure 8: PSNR(dB) comparison

6 Conclusion

We propose a novel convolutional neural network DE-CNN for pixel-wise depth image denoising and enhancement [?]. From the experiments above, the model has high performance and high computational efficiency. This model can be applied towards real world image processing applications.

7 Reference

1. Microsoft, Kinect,.

2. Intel, Realsense,.

3. N. Fukushima, H. Jin, R., and Y. Ishibashi, Weighted joint bilateral filter with slope depth compensation filter for depth map refinement, in International Conference on Computer Vision

Theory and Applications, 2011, pp.300309.

4. M. Wang, T. Matsuo, Yang, and M. Gong, Stereoscopic inpainting: Joint color and depth completion from stereo images, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 18.

5. J. Shen and S. C. S. Cheung, “Layer Depth Denoising and Completion for Structured-Light RGB-D Cameras,” IEEE Conference on Computer Vision and Pattern Recognition, pp. 1187-1194, 2013.

6. L. Fu, S. Zhao, Y. Lu, and W. Zeng, Kinect-like depth denoising, in IEEE International Symposium on Circuits and Systems, 2014, pp. 512515.

7. W. Wang and F. F. Cai, Data-driven tight frame for multi-channel images and its application to joint colordepth image reconstruction, Journal of the Operations Research Society of China, vol. 3, no. 2, pp. 99115, 2013.

8. J. Shen and J. Yang, “Automatic human animation for non-humanoid 3d characters,” International Conference on Computer-Aided Design and Computer Graphics (CAD/Graphics), pp. 220-221, 2015.

9. R. Lu, Xiaofeng Ren, and Feng Liu, Depth enhancementc via low-rank matrix completion, in Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 33903397.

10. T. Rinton, P. Dayan, B. J. Frey, and R. M. Neal, The wake-sleep algorithm for unsupervised neural networks, Science, pp. 11581161, 1995.

11. Z. Sun, X. Wang, and X. Tang, Deep convolutional network cascade for facial point detection, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 34763483.

12. J. Shen, P. C. Su, S. c. S. Cheung, J. Zhao, “Virtual mirror rendering with stationary rgb-d cameras and stored 3-d background,” IEEE Transactions on Image Processing, vol. 22, no. 9, pp. 3433-3448, 2013.

13. D. C. Loy, K. He, and X. Tang, Learning a deep convolutional network for image super resolution, in Proceedings of Eurpoean Conference on Computer Vision, 2012, pp. 184199.

14. L. Long, E. Shelhammer, and T. Darrell, Fully convlutional networks for sematic segmentation, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2017.

15. J. Shen and W. Tan, “Image-based indoor place-finder using image to plane matching,” IEEE International Conference on Multimedia and Expo, pp. 1-6, 2013.

16. D. Scharstein and R. Szeiliski, High-accuracy stereo depth maps using structured light, in Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2007, pp. 195202.

17. E. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, PatchMatch: A randomized correspondence algorithm for structural image editing, in ACM Transactions on Graphics, 2011.

18. FAST DEPTH IMAGE DENOISING AND ENHANCEMENT USING A DEEP CONVOLUTIONAL NETWORK Yin Wang and Yuan Wang, 2014.