Noise Reduction of Depth Camera Images using Deep Neural Networks

Seyed Mehrdad Mahdavi[1]*, Mohsen Ashourian[2]
1- Department of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran.
Email: mehrdad5087@gmail.com (Corresponding author)
2- Department of Electrical Engineering, Majlesi Branch, Islamic Azad University, Majlesi, Iran.
Email: ashourian@iaumajlesi.ac.ir

Received: February 2020 Revised: June 2020 Accepted: July 2020

ABSTRACT:
Today, infrared sensors or depth sensors are widely used to control applications, games, information acquisition, dynamic and static 3D scenes. Despite the widespread use of these images, their quality is limited to low-quality images, as the infrared sensor does not have high resolution and the images produced by it have noise. Therefore, given the problems and the importance of using 3-D images, the quality of these images should be improved in order to provide accurate images from depth cameras. In this paper, the noise reduction of depth images using convolutional neural networks is considered. A convolutional neural network with a depth of 20 and three layers and a pre-trained neural network is used. We developed the system and tested its performance for two datasets of depth and color images, Middlebury and EURECOM Kinect Face. Results show that for EURECOM Kinect Face images, PSNR improvement is approximately 8 to 15 dB and for Middlebury images the PSNR improvement is about 5 to 14 dB.

KEYWORDS: Depth Camera Images, Image Enhancement, Noise Reduction, Convolution Neural Networks.

1. INTRODUCTION

Depth cameras consist of a standard color camera and an infrared camera. The color camera captures a color image, as the name implies, from the environment and scene, which is used in subsequent processing to improve the depth of field images, and the depth camera estimates the depth by reflecting infrared light. In these cameras, a light source transmits infrared light with a dot pattern, then the sensors, which are the heart of the camera, take a recursive pattern and estimate the distance based on the length of sweet time of the light. Fig. 1 shows how to estimate the depth.

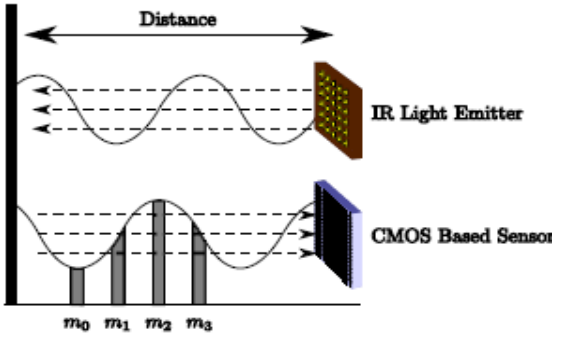

Fig. 1. How to estimate depth in depth cameras.

Three-dimensional information from a scene includes position information (depth information) and texture information. While texture information is easily captured by color cameras, it is not easy to obtain depth information. In addition, the obtained depth information require pre-processing to be used for subsequent processing, (Improvements), since, the depth maps recorded by the depth cameras have very low resolution compared to the color image. These depth maps also suffer from various damages such as low sampling, loss of structure along the depth discontinuities, and accidental loss in smooth areas, which make these images noisy with lack of sharpness and ultimately their quality would be reduced. Such destructions have hampered their practical application. Depth cameras have errors when shooting objects with special features such as sharp edges, and their error increases in very bright environments. These problems are caused by changes in ambient brightness, scene geometry, ambient heat, and elevated sensor temperatures over time, so given the current problems and the wide usage of infrared sensors to control application and games and obtain information from dynamic and 3-D scenes, the image quality of these cameras has to be improved. Despite the widespread use of these images, their quality is limited to low-quality and noisy images, because the infrared sensor does not have high resolution and the

Doi: https://doi.org/10.29252/mjee.14.3.12
How to cite this paper: S.M. Mahdavi, M. Ashourian, **"Noise Reduction of Depth Camera Images using Deep Neural Networks"**, *Majlesi Journal of Electrical Engineering*, Vol.14, No. 3, pp. 95-100, 2020.

images produced by it have noise. Therefore, due to the existing problems and the importance of using depth images, the quality of these images should be improved in order to provide accurate images using depth cameras [2].

In the last few years, due to the widespread use of these cameras, a great deal of research has been done to improve the depth images quality, all of which try to improve the images both in terms of noise reduction and sharpness. However, at first, attempts to improve depth images have used only laser sensors or only the camera, and combining these two methods is a relatively new topic. In this paper, we try to reduce the noise of depth images by using depth estimation method by neural network model.

2. LITERATURE REVIEW

Noise reduction due to its importance and applications in various fields is still one of the hot topics in the field of machine vision. The main idea of noise reduction is to extract the clear image of x from the noise image of y which is as y= x + v. The common assumption in this case is that they assume that v is the cumulative noise by Gaussian distribution with the standard deviation of σ [3]. Different models for the original image have been used in papers. In [4–8], they used a self-similarity model for the original image to estimate the original image through different algorithms, and their evaluation criterion is the mean square error between the resulting image and the original image. In these articles, the authors present a Non-local Self-Similarity (NSS) algorithm and try to minimize the estimated image error and the original image, thereby reducing or eliminating noise from the image. They use pixel-sized neighborhood windows of different sizes for this averaging. These methods, although have reasonable result, but have two major disadvantages: first, they deal with sophisticated optimization calculations that lead to time-consuming and prolonged execution and, on the other hand, they do not perform well without solving the optimization problem. In addition, their second disadvantage is their non-convexity, which makes their performance dependent on the selection of their parameters. Therefore, in order to overcome these two flaws, discriminative learning methods have been introduced [9]. This learning method is used instead of solving the optimization problem to reduce image noise. Although they are able to compensate for the gap between computational burden and noise removal quality, the method itself was also dependent on the initial model of the image, which does not seem desirable.

Based on the above difficulties, recently researchers are using Convolutional Neural Networks (CNNs) instead of realistic modeling learn from the original image. There are three main reasons for using these types of networks:

• Convolutional neural networks with deep architecture have the capability and flexibility to describe image properties [10].

• Another notable advantage of these types of networks is their learning methods, which include Rectifier Linear Unit (ReLU) [11], batch normalization [12], and residual learning [13]. In these papers, this type of learning is introduced for classification and recognition tasks, but can also be used as a future research area to reduce noise and speed up the learning process of the network.

• Convolutional neural networks using parallel computing are well compatible with modern, powerful GPUs, which can be used to reduce their running time.

The most important reason for using CNN to reduce image noise is that it does not require to estimate original image, and the noise is estimated directly. This is done by the difference between the noisy image and the clear noise. It should be noted that this paper uses color information-based estimators because if only depth sensors are used, the depth results depend on how they are navigated and for these results to be accurate and produce high quality images, very good navigation should be performed on them [14, 15]. Also, if a depth sensor is used multiple times instead of using multiple depth sensors, if the environment changes during the use of the sensor, the fusion of the results would become problematic [16, 17]. To overcome these problems, a color camera that produces a high-quality color image can be used to improve the quality of the low-quality depth image produced by the depth sensor. In fact, a color image is used to take advantage of adjacent of the dots in the color image and the associated depth image, increasing their ability to measure local similarities, turning them from piecewise into patches and having less processing complexity than other methods.

3. RESEARCH METHODOLOGY

In terms of network architecture design, our proposed method is a modified VGG network [10]. VGG can be used to reduce image noise. In the proposed method the depth of network is adjusted based on patch sizes. In terms of model learning, the residual learning formulation has also been selected and combined with patch normalization to accelerate training and improve noise reduction performance. Similar to the method used in Reference [10], in this paper, the size of the convolutional filters is assumed 3×3, except that all pooling layers are removed. In the proposed convolutional neural network architecture, the observed noisy image of $y = x + v$ is the input the network where x is the original image and υ is the additive noise. The image can have any dimension, or it can even be gray or colored. In [18], noise removal models are considered as

the function of $F(y)=x$ and by using this function, they can predict and estimate the clear and noise-free image. However, in our proposed convolutional neural network approach, the residual learning formulation is used to train a residual mapping like $R(y)\approx v$. In other words, unlike other articles that estimate the clear and noise-free image, we estimate noise that is assumed to be of an unknown nature. Then, with the estimated noise, we can easily obtain a clear, noise-free image through $x=y-R(y)$. For this purpose, the mean square error criterion is used, but it should be noted here that the error refers to the difference between the desired image and the estimated image and has the following relation (1):

$$l(\theta)=\frac{1}{2N}\sum_{i=1}^{N}\left\|R(y,\theta)-(y_i-x_i)\right\|_F^2 \quad (1)$$

This function can be used as a target function to learn the training parameters of θ available in convolutional neural networks. Here, $\{(y_i-x_i)\}_{i=1}^{N}$ represents N images of the noisy and clear training which together, they make a pair of patches. This function is used to train residual mapping of $R(y)\approx v$ and then, $x=y-R(y)$ can be easily extracted. According to the mean square theory, the error between the desired residual images and the estimated image of the noisy input can be used as an error function to learn the parameters of θ, and therefore v that is unknown can be estimated. Fig. 2 shows the proposed neural network architecture for learning.

Fig. 2. Proposed convolutional neural network architecture.

As can be seen from Fig. 2, the noisy image from the left enters the system, and on the right, the residual image is obtained. It is evident from the figure that there are three layers in the convolutional neural network that are shown in different colors. These three layers are as follows:

1. Conv + ReLU: This layer is the first layer of convolutional neural network shown in yellow. This layer contains 64 filters with a size of $3\times3\times c$, whose task is to create 64 feature mappings; and there are also 64 rectifier linear units (ReLU) that provide nonlinearity. The parameter c represents the image type or number of channels of the image, so if $c=1$, the image is gray and if $c=3$, the image is colored.
2. Conv + BN + ReLU: These layers are in the second layer to the $D-1$ layer and are shown in blue. In this layer, 64 filters with a size of $3\times3\times64$ are used, as well as patch normalization introduced in Reference [12] between convolution and ReLU.
3. Conv: This is the last layer in orange (Fig. 2). It has a number of c filters with a size of $3\times3\times64$ used to form the output.

There is something about the size of the input and output image that needs to be mentioned here. In many low-level machine vision applications, the output image and the input image need to be equal in size. Some references in the final stage consider a process to reduce the size of the image and make it as the size of input image [18]. In the proposed method, before the image reaches the final layer or the convolution layer, it is ensured by using a condition that the output image and the input image are equal and the work of equalizing them is done in the convolution layer.

4. RESULTS AND DISCUSSION

The proposed method has been applied to different images with different noise levels. For this purpose, two types of datasets are considered, which are introduced below:

1-EURECOM Kinect Face dataset: This dataset is about the faces of different people who were photographed using a Kinect camera. In this dataset, 52 people are photographed at 9 different angles and their depth and

color image are collected. [19]. An example of noisy images of the EURECOM Kinect Face dataset is shown in Fig. 3.

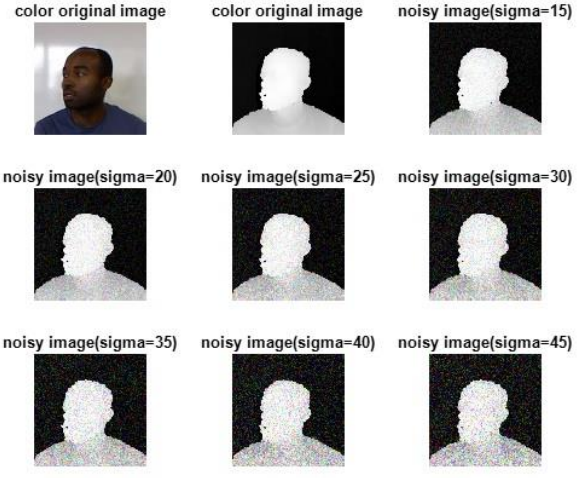

Fig. 3. An example of the EURECOM Kinect Face dataset.

2-The second dataset is known as Middlebury, in which the collection contains images of various sights and scenes. This dataset was compiled in 2001, 2003, 2005, 2006 and 2014 [20-23]. An example of noisy images of the Middlebury dataset is shown in Fig. 4.

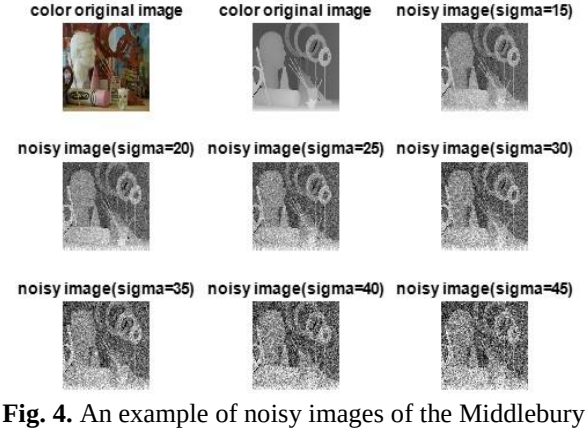

Fig. 4. An example of noisy images of the Middlebury dataset.

In this simulation, a pre-trained convolutional neural network with 400 images and the learning method mentioned in the reference [24] are used for training, all images have dimensions of 180*180. The noise level is assumed to be unknown and belongs to the range of $\sigma \in [0,55]$. In this case, the size of the used patches is assumed to be 50*50, so that the total number of 128*3000 patches were used to train the noise removal model. The number of network depths is 20 and the loss function is (1) selected to learn the network. The weights of the network were calculated by the method used in the reference [25] and the gradient descent method with a weight delay of 0.0001. In addition, network training over 50 iPOCs has been able to build noise removal models.

It should be noted that the MatConvNet [26] library has been used for network training. The simulations are performed in MATLAB software. After running, a convolutional network with 33 layers is created which each layer's information is stored in a structure. This information includes layer type, layer weights, weight delay, learning rate, and so on.

Middlebury dataset results: Art image of Middlebury dataset with 5, 10, 15, 20, 25, 30, 35, 40 noise levels was impregnated with Gaussian noise as inputs into the convolutional neural network and the noise eliminated image exits the network. Three examples of this are shown in Fig. 5 to Fig.7.

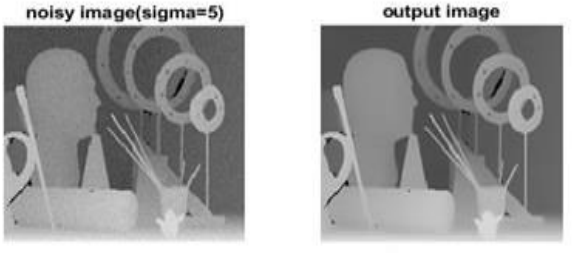

Fig. 5. Art image from Middlebury dataset: Left: Noisy image with $\sigma = 5$. Right: Network output image.

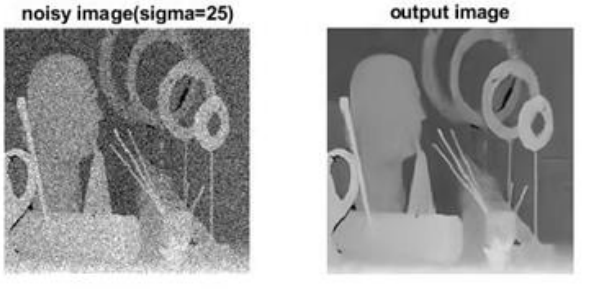

Fig. 6. Art image from Middlebury dataset: Left: Noisy image with $\sigma = 25$. Right: Network output image.

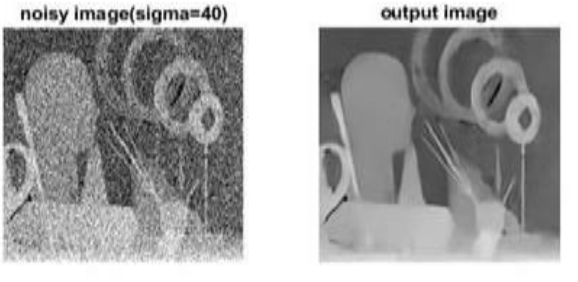

Fig. 7. Art image from Middlebury dataset: Left: Noisy image with $\sigma = 40$. Right: Network output image.

EURECOM Kinect Face Datasheet Results: For this datasheet, an image with the above noise levels was impregnated with the Gaussian noise and enters the convolutional neural network as an input, and the noise eliminated image exits from the network, as shown in Fig. 8 to Fig. 10.

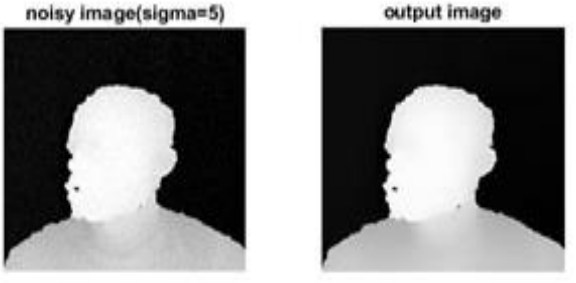

Fig. 8. A sample image of the EURECOM Kinect Face dataset: Left: Noisy image with $\sigma = 5$. Right: Network output image.

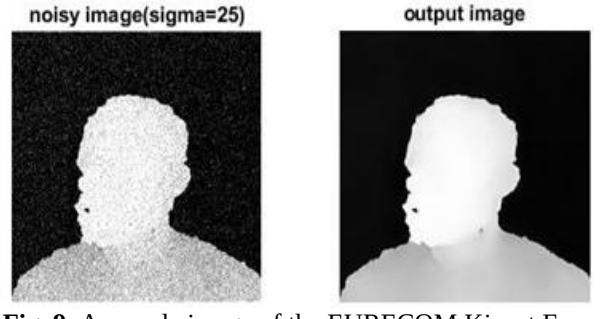

Fig. 9. A sample image of the EURECOM Kinect Face dataset: Left: Noisy image with $\sigma = 25$. Right: Network output image.

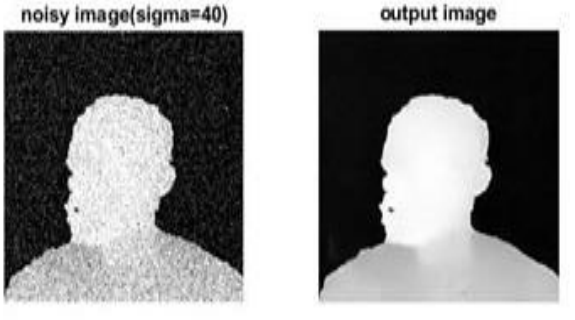

Fig. 10. A sample image of the EURECOM Kinect Face dataset: Left: Noisy image with $\sigma = 40$. Right: Network output image.

In Table 1, the result of the PSNR calculations of the noisy input depth image and the PSNR of depth image obtained from the network corresponding to the image sample of the two datasets for the noise levels of 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 is provided.

Table 1. Results of PSNR calculations of noisy input depth image and PSNR depth image obtained from the network.

σ	dataset EURECOM Kinect Face		Middlebury dataset	
	PSNR input	PSNR output	PSNR input	PSNR output
$\sigma = 5$	346913	420918	341626	391201
$\sigma = 10$	293063	413961	280911	383803
$\sigma = 15$	261889	392579	245918	361287
$\sigma = 20$	238089	370331	221095	341708
$\sigma = 25$	221449	358703	201593	326872
$\sigma = 30$	206064	346694	186401	319675
$\sigma = 35$	193841	340378	176367	310778
$\sigma = 40$	183850	333708	162719	296686
$\sigma = 45$	173678	327613	153514	292933
$\sigma = 50$	165459	315957	145071	284860

5. CONCLUSION

In this paper, a noise removal model for depth images is presented with the use of convolutional neural networks and information obtained from color and depth image. The proposed model is applied to two Middlebury and EURECOM Kinect Face datasets with different noise levels and it was shown that the output PSNR has higher values than the input and has improved. It is assumed in this study that there are Gaussian noise in deep images and only the Gaussian noise is eliminated, while other types of noise can be investigated. In addition, the created convolutional neural network can be expanded so that it can eliminate other types of noise as well.

REFERENCES

[1] S. Foix, G. Alenya, and C. Torras, **"Lock-in time-of-flight (ToF) cameras: A survey,"** *IEEE Sensors Journal*, Vol. 11, No. 9, pp. 1917-1926, 2011.

[2] D. Csetverikov, I. Eichhardt, and Z. Jankó, **"A brief survey of image-based depth upsampling,"** 2015.

[3] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, **"Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising,"** *IEEE Transactions on Image Processing*, Vol. 26, No. 7, pp. 3142-3155, 2017.

[4] A. Buades, B. Coll, and J.-M. Morel, **"A non-local algorithm for image denoising,"** in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, Vol. 2, pp. 60-65: IEEE.

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, **"Image denoising by sparse 3-D transform-domain collaborative filtering,"** *IEEE Transactions on image processing*, Vol. 16, No. 8, pp. 2080-2095, 2007.

[6] A. Buades, B. Coll, and J.-M. Morel, **"Nonlocal image and movie denoising,"** *International journal*

of computer vision, Vol. 76, No. 2 ,pp. 123-139, 2008.

[7] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, **"Non-local sparse models for image restoration,"** in *Computer Vision, 2009 IEEE 12th International Conference on*, 2009, pp. 2272-2279: IEEE.

[8] J. Xu, L. Zhang, W. Zuo, D. Zhang, and X. Feng, **"Patch group based nonlocal self-similarity prior learning for image denoising,"** in *Proceedings of the IEEE international conference on computer vision*, pp. 244-252, 2015.

[9] U. Schmidt and S. Roth, **"Shrinkage fields for effective image restoration,"** in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2774-2781, 2014.

[10] K. Simonyan and A. Zisserman, **"Very deep convolutional networks for large-scale image recognition,"** *arXiv preprint arXiv:1409.1556* 2014.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, **"Imagenet classification with deep convolutional neural networks,"** in *Advances in neural information processing systems*, pp. 1097-1105, 2012.

[12] S. Ioffe and C. Szegedy, **"Batch normalization: Accelerating deep network training by reducing internal covariate shift,"** *arXiv preprint arXiv:1502.03167*, 2015.

[13] K. He, X. Zhang, S. Ren, and J. Sun, **"Deep residual learning for image recognition,"** in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.

[14] S. A. Gudmundsson, H. Aanaes, and R. Larsen, **"Fusion of stereo vision and time-of-flight imaging for improved 3d estimation,"** *International Journal of Intelligent Systems Technologies and Applications*, Vol. 5, No. 3-4, pp. 425-433, 2008.

[15] J. Zhu, L. Wang, R. Yang, and J. E. Davis, **"Reliability fusion of time-of-flight depth and stereo geometry for high quality depth maps,"** *IEEE transactions on pattern analysis and machine intelligence*, Vol. 33, No. 7 ,pp. 1400-1414, 2011.

[16] B.-S. Lin, W.-R. Chou, C. Yu, P.-H. Cheng, P.-J. Tseng, and S.-J. Chen, **"An effective spatial-temporal denoising approach for depth images,"** in *2015 IEEE International Conference on Digital Signal Processing (DSP)*, 2015, pp. 64-67, IEEE.

[17] S. MJ, **"Temporal and Spatial Denoising of Depth Maps,"** *Sensors (Basel). 2015 Jul 29*, No. 8, 2015.

[18] H. C. Burger, C. J. Schuler, and S. Harmeling, **"Image denoising: Can plain neural networks compete with BM3D?,"** in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 2392-2399: IEEE.

[19] R. Min, N. Kose, and J.-L. Dugelay, **"Kinectfacedb: A kinect database for face recognition,"** *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 44, No. 11 ,pp. 1534-1548, 2014.

[20] D. Scharstein and R. Szeliski, **"A taxonomy and evaluation of dense two-frame stereo correspondence algorithms,"** *International journal of computer vision*, Vol. 47, No. 1-3, pp. 7-42, 2002.

[21] D. Scharstein and R. Szeliski, **"High-accuracy stereo depth maps using structured light,"** in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, 2003, Vol. 1, pp. I-I: IEEE.

[22] D. Scharstein and C. Pal, **"Learning conditional random fields for stereo,"** in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 2007, pp. 1-8: IEEE.

[23] H. Hirschmuller and D. Scharstein, **"Evaluation of cost functions for stereo matching,"** in *Computer Vision and Pattern Recognition, 2007 .CVPR'07. IEEE Conference on*, 2007, pp. 1-8: IEEE.

[24] Y. Chen and T. Pock, **"Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration,"** *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39, No. 6, pp. 1256-1272, 2017.

[25] K. He, X. Zhang, S. Ren, and J. Sun, **"Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,"** in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026-1034.

[26] A. Vedaldi and K. Lenc, **"Matconvnet: Convolutional neural networks for matlab,"** in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 689-692: ACM.