Kinect Depth Recovery via the Cooperative Profit Random Forest Algorithm

Jianyuan Sun[†,††], Qi Lin[†,*], Xuguang Zhang[†††], Junyu Dong[†,*], Hui Yu[††]
[†]Department of Computer Science and Technology
Ocean University of China, Qingdao, China
[†††]School of Communication Engineering
Hangzhou Dianzi University, Hangzhou, China
[††]School of Creative Technologies
University of Portsmouth, Portsmouth, United Kingdom
[*]Corresponding author (email: dongjunyu@ouc.edu.cn; qilin@ouc.edu.cn)

***Abstract*—The depth map captured by Kinect usually contain missing depth data. In this paper, we propose a novel method to recover the missing depth data with the guidance of depth information of each neighborhood pixel. In the proposed framework, a self-taught mechanism and a cooperative profit random forest (CPRF) algorithm are combined to predict the missing depth data based on the existing depth data and the corresponding RGB image. The proposed method can overcome the defects of the traditional methods which is prone to producing artifact or blur on the edge of objects. The experimental results on the Berkeley 3-D Object Dataset (B3DO) and the Middlebury benchmark dataset show that the proposed method outperforms the existing method for the recovery of the missing depth data. In particular, it has a good effect on maintaining the geometry of objects.**

***Index Terms*—depth map, neighborhood pixel, cooperative profit random forests, missing depth data**

I. Introduction

Microsoft Kinect sensor is very popular in the domain of smart human-computer interaction due to less interference under the illumination change and the complex background situation [1–3]. Moreover, it can capture both the depth map and the corresponding RGB map for a wide range of scenes. The captured depth map and the corresponding RGB map can be used to extract the characteristics of human actions or hand motion, which are crucial elements for human-computer interaction [4]. However, it is generally known that the captured depth map usually comes with missing depth data at the boundary of objects or the surface of infrared absorption. A typical example is given in Fig. 1. Therefore, filling hole (missing depth data) becomes an essential preprocessing step.

Most of existing methods on the recovery depth map mainly use some filtering methods. Such as the median filter, the joint bilateral filter and the guided filter. However, these methods are prone to producing artifact or blur on the large area of missing depth data. In order to achieve good visual effects, the captured RGB map is used to predict missing depth data. We can learn a lot of useful information from the RGB images, such as texture information, the geometry of objects and spatial information. Therefore, the texture-assisted and image inpainting techniques are developed to recover missing depth values. Unfortunately, these approaches cannot obtain satisfactory results for the region of depth discontinuity. Recently, there are some methods based on the combination of the traditional methods and the machine learning methods. However, most of these methods have great limitations in the predicted scenes.

978-1-5386-5024-0/18/$31.00 ©2018 IEEE

Fig. 1. (a) and (b) are the RGB image and the corresponding depth map, which are captured by Microsoft Kinect. (c) is the restore result by using the proposed method.

To deal with the current Kinect depth recovery issues in a unified framework, we propose a novel method to recover the missing depth data with the guidance of depth information of each neighborhood pixel. In this framework, a cooperative profit random forest (CPRF) algorithm is used to predict the missing depth data based on the existing depth data and the corresponding color map. In particular, a self-taught learning mechanism combines the CPRF to predict the missing depth data. The CPRF can explore the interdependency relation between attributes (pixels) for a learning task (depth information) [5]. Therefore, the proposed algorithm works well for restoring the edge structure of the object and considering the spatial structure of the objects in the scene at the same time. The experimental results on the Berkeley 3-D Object Dataset (B3DO) and the Middlebury dataset show that the proposed method outperforms the existing method for the recovery of the missing depth data.

The rest of the paper is organized as follows. Section 2 reviews the progress of the recovery depth map in recent years.

Section 3 describes the proposed method framework in detail. In Section 4, the experiment results and the corresponding result analysis are presented. Section 5 concludes this work.

II. Related Work

Most existing mainstream methods for recovering Kinect depth maps mainly are use the filtering methods. The simplest method hole filling is to apply median filter [6] to the depth and RGB image (RGB-D). However, using the median filter method often brings several blurs onto object edges. To overcome this limitation, the joint bilateral filter [7] is employed to fill holes in the RGB-D. However, when the holes are too large, this operation is easy to produce the artifact. In addition, the guided filter [8] is used to maintain sharp edges and avoid artifacts for restoring the depth map from Kinect. However, these filter-based methods do not adequately consider the large dark holes in the Kinect depth maps and are prone to producing blur on the object edge. Therefore, the texture-assisted scheme [9] and image inpainting techniques [10] are applied and developed to restore the missing depth values. But these approaches cannot obtain satisfactory results for the regions of depth discontinuities.

In recent years, some novel methods are proposed based on the benefit of the strong correlation between the depth maps and the associated RGB images. These novel methods tend to combine the traditional depth recovery methods and the machine learning methods. For example, a graph Laplacian based framework is proposed [11] to recover the missing depth information (i.e., holes). In addition, an effective method combines an existing auto-regressive model [12] and a new filter method. Moreover, a self-taught regression method is proposed to restore the missing depth data [13]. In particular, the original random forests regression algorithm [14] is employed to predict depth data. This method is not only based on the strong correlation between the depth maps and the associated RGB images but also combines an initial rough estimation of the depth. The Make 3D [15] technology is used to obtain the rough depth estimation. However, this method has its limitations. It is only valid for the objects of tables and chairs in the indoor scene. In addition, the node split method of the original random forests regression algorithm often pays less attention to the intrinsic structure of the attribute (pixel) variables and tends to ignore attributes (pixels) with strong discriminate ability as a group yet weak as individuals [5].

In this paper, we propose a simple and efficient framework for the depth map recovery. In particular, to achieve high-quality depth recovery, we use the cooperative profit random forests (CPRF) classification algorithm to predict the missing depth information [5]. The construction of CPRF is based on the cooperative game, which uses the Banzhaf power index to expand tree nodes. The split criterion of Banzhaf power index in CPRF can explore the dependency relation of attributes (pixels) for learning objective. Moreover, we combine neighborhood information of the gray-scale intensity of RGB image and the depth map captured by Kinect to train CPRF. In particular, we specify that the pixels with the largest number of the valid neighborhood are predicted first and then predicted depth information is added to the training set for the next round of prediction.

III. Proposed Method

This section describes the cooperative profit random forests (CPRF) formulation of the Kinect depth recovery, which takes the benefit of the strong correlation between the Kinect depth maps and the corresponding RGB images. We define the symbol D_k and I_c to represent the Kinect depth map and the corresponding RGB map respectively. Then, the pixel value of the missing depth information is 0 in D_k. Inspired by the work of Yang *et al.* [13], it can be as the following equation.

$$\hat{D}_k = \{D_k(i,j)|D_k(i,j) = 0\}.$$

We train the CPRF based on the neighborhood information of I_c and D_k. In particular, CPRF predicts the missing depth data in a multi-round, and the pixels with the largest number of the valid neighborhood are predicted first. Then the predicted depth values are added to the training set for the next round prediction. Repeat this way until all missing depth data is filled. The flow graph of our framework is shown in Fig. 2.

Fig. 2. The proposed framework.

A. Cooperative profit random forests (CPRF)

The cooperative profit random forests algorithm (CPRF) is a classification tool [5]. CPRF is an ensemble algorithm, which combines several Profit decision trees (PDTs). The final prediction results of CPRF are based on the majority votes among the PDTs. In particular, each PDTs employs the Banzhaf power index as the node split criterion to evaluate the best split point and the corresponding feature at each tree node. As described in the work [5], the split criterion of

Algorithm 1. Profit Decision Tree (PDT)

1: **Initialize**: Given the training dataset $D = \{(x_1, y_1), \ldots, (x_n, y_n)\} \in R^{n\times(p+1)}$, and the feature variables $f_j = (x_{1,j}, \ldots, x_{n,j})^T$, $j = (1, \ldots, p)$, $T = \varnothing$, $\epsilon = 0$, let $B_{root} := D$;
2: TreeBlock(root, B_{root})

Algorithm 1.1 TreeBlock (f_j, B_{f_j})

1: Add f_j to T, $j \in (1, \ldots, p)$
While
2: Symbol $\epsilon_{i,j}$ denotes the split threshold of the feature f_j, $\epsilon_{i,j} = (x_{i,j} + x_{i,j+1})/2, i = 1, \ldots, n$, $j \in (1, \ldots, p-1)$;
3: For the split threshold $\epsilon_{i,j}$ of each feature f_j $(j = 1, \ldots, p)$ do;
4: Calculate: $\gamma(NL) = \sum_{j=1}^{p} \eta(f_j), f_j \in B_{left(f_j)}$, $\gamma(NR) = \sum_{j=1}^{p} \eta(f_j), f_j \in B_{right(f_j)}$, where $B_{left(f_j)} = \{(f_1, \ldots, f_p) \in B_{f_j} : x_{i,j} \le \epsilon_{i,j}\}$, $B_{right(f_j)} = \{(f_1, \ldots, f_p) \in B_{f_j} : x_{i,j} > \epsilon_{i,j}\}$ and $\eta(f_j)$ is the Banzhaf power index (gains) of each feature f_j $(j = 1, \ldots, n)$;
5: $Split(\hat{f}_j, \epsilon_{\hat{i},j}) \leftarrow \arg\max(\gamma(NL) + \gamma(NR))$, set $B_{left(\hat{f}_j)} = \{(f_1, \ldots, f_p) \in B_{\hat{f}_j} : x_{i,j} \le \hat{\epsilon}_{i,j}\}$ and $B_{right(\hat{f}_j)} = \{(f_1, \ldots, f_p) \in B_{\hat{f}_j} : x_{i,j} > \hat{\epsilon}_{i,j}\}$;
6: TreeBlock(left($\hat{f}_j$), $B_{left(\hat{f}_j)}$)
7: TreeBlock(right($\hat{f}_j$), $B_{right(\hat{f}_j)}$)
Until reaching the user-set limit, i.e., a minimal number of samples of a node.
8: f_j to be the $leaves(T)$

Banzhaf power index can be learned the internal relationships between features variables, i.e., a group of feature variables. It has a strong discrimination ability in term of the target class, and can learned in each PDTs. The construction of PDT is described in Algorithm 1. Moreover, the calculation of the Banzhaf power index for each feature can refer to the work [5] in details.

For the depth recovery problem, the majority of the methods view it as a regression problem [16, 17]. However, we think that it is more appropriate to regard it as a classification problem since the pixel values of the Kinect depth map are positive integers and the pixel values of the local large area are the same in the Kinect depth map, such as the background pixels and the pixels of a component of the object, as shown in Fig. 3. Therefore, we take the 8-neighborhood of the valid depth information and 8-neighborhood of corresponding gray intensity of the corresponding RGB image to form a training dataset. Then, we only need to learn and predict a small number of categories. Furthermore, the experimental results verify the effectiveness of the proposed method.

B. The depth recovery method

Microsoft Kinect can supply an RGB image and its aligned depth map simultaneously. In a 3D scene, different objects are made up of different pixel values, and the neighboring pixels of the same object often share the same or similar properties. Inspired by this observation and the

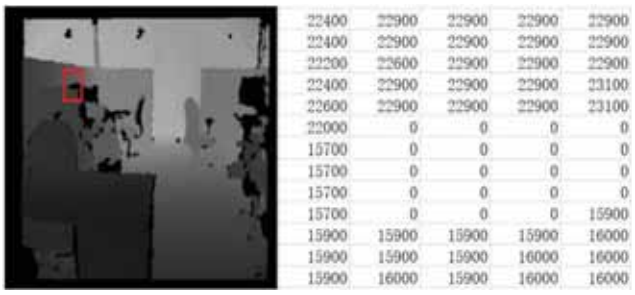

Fig. 3. The image on the right shows some pixel values of the red frame area on the left depth map.

work of Yang *et al.* [13], we use neighborhood information of the depth image and the corresponding RGB image to predict the missing depth values. In particular, the 8-neighborhood of each pixel is used. $N_{i,j}$ denotes the 8-neighborhood of the pixel (x, y) at image location (i, j), as follows

$$N_{i,j} = \{(x, y) | i - 1 \le x \le i + 1, j - 1 \le y \le j + 1\}.$$

Then the training dataset can be obtained, i.e. for the depth map, the valid depth value itself as the training label, the 8-neighborhood of the valid depth value and the corresponding RGB image′s pixel value as the data features of the training set.

$$T_{train} = \{N^*_{i,j} | D_k(i, j) \notin \hat{D}_k, |N^*_{i,j}| = 8\}.$$

For the test dataset, the 8-neighborhood of the non-valid (missing) depth value and the corresponding values of the RGB image as dataset features. Then the Cooperative Profit Random Forests algorithm (CPRF) is employed to predict the non-valid (missing) depth value. To ensure the accuracy of prediction, we first predict these missing pixels where the valid neighborhood pixels values are equal to or greater than 4. After restoring the pixels of the missing depth values, the training set T_{train} is updated by adding the estimated depth values. The CPRF is re-trained on the updated training dataset, and the re-trained CPRF model is used for the next round prediction. Repeating this procedure until all missing depth of the depth image are restored. The proposed method is similar to the self-taught learning method. Self-taught learning is a popular learning method in machine learning, which has been used in depth-in-painting [13], classification applications [18, 19] and image retrieval [20]. In our work, the proposed depth image recovery method is called self-taught classification algorithm (STC).

The complete depth recovery method is described in Algorithm 2.

IV. Experiments and discussion

In this section, to verify the effectiveness of the self-taught classification depth recovery algorithm, we implement the contrast experiment based on the Berkeley 3-D Object Dataset (B3DO) and the Middlebury benchmark dataset.

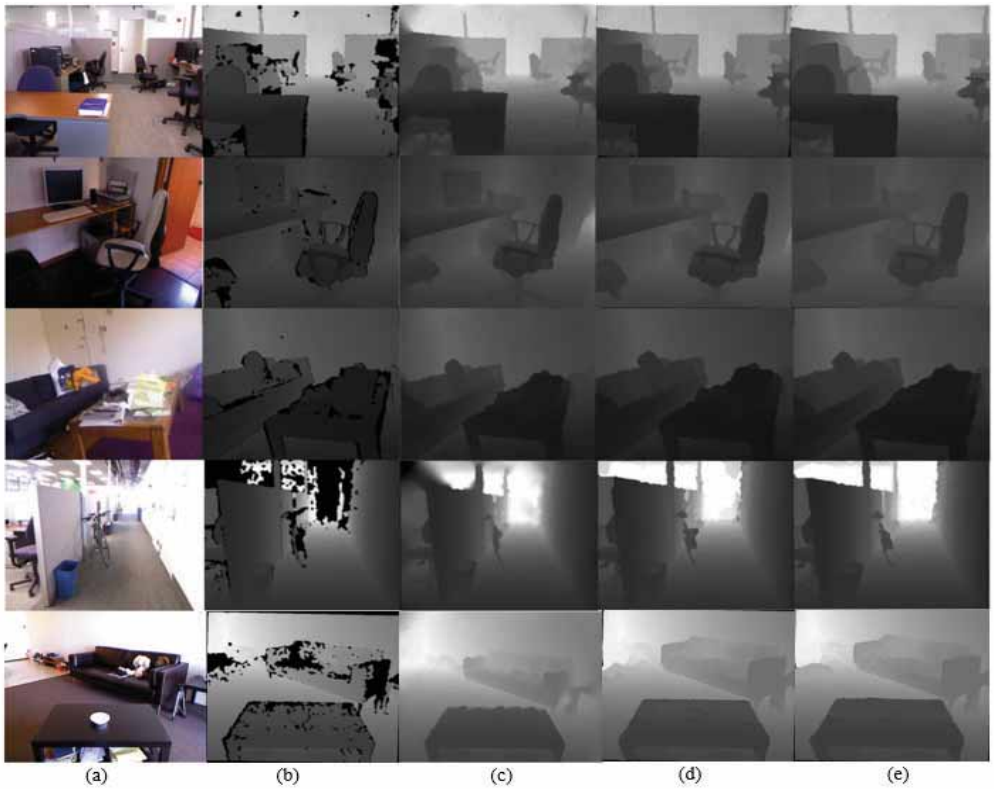

Fig. 4. Restored depth image on B3DO dataset. (a) RGB image. (b) Incomplete depth image. (c) Restored depth image provided by B3DO dataset. (d) Restored depth image by self-taught regression algorithm [13]. (e) Our method.

Algorithm 2: The self-taught classification algorithm (STC) for the Kinect depth recovery

Initialize: RGB image I_c and Depth image D_k are captured by Kinect respectively.
For: I_c and D_k
1: Collection training dataset: the 8-neighborhood of valid depth value of D_k and the corresponding RGB images pixel value of I_c as dataset features to form training dataset T_{train};
2: Collection test dataset: the 8-neighborhood of non-valid (missing) depth value and the corresponding RGB images pixel value as dataset features to form test dataset T_{test};
3: Training Cooperative Profit Random Forests method (CPRF) based on T_{train};
4: Based on T_{test} , using a trained CPRF mode to predict the missing depth value that has the largest number of valid neighboring pixel;
5: **Return** D_k;
6: Update T_{train} according to the result of step 4;
7: Repeat step 3, 4, 5, 6;
Until $T_{test} = \varnothing$.

A. The Berkeley 3-D Object Dataset (B3DO)

The B3DO dataset consists of RGB images and the corresponding depth images of missing depth information, which are mainly office scenes. In addition, the B3DO dataset provides the restored depth image by applying average filter methods or using a global descriptor, which will be used to compare our method. Furthermore, an existing self-taught depth regression recovery method [13] is also used to compare with our depth recovery method. The existing self-taught depth recovery method is similar to our method, but the major difference is that the self-taught depth recovery method fuses a rough scene depth estimation obtained by Make 3D [15] and uses the original random regression forests to predict missing depth information.

The experimental results are shown in Fig. 4. From the results, we can observe that the restored depth image provided by B3DO dataset that tends to produce blur at the edge of an object; the self-taught regression algorithm [13] performs better than the restored depth image from B3DO dataset. However, compared with our method, it is easy to see that our method is more accurate than the self-taught regression algorithm [13] in term of detail restoration of objects. The details of the recovered objects are shown in Fig. 5.

Results analysis The B3DO dataset provides the restored depth image by applying an average filter method or using global descriptors. Some filter methods or global descriptor are prone to bringing blur to the edges of objects in the task of depth image inpainting, which has been proved by previous works [6–8]. The self-taught regression algorithm [13] and our algorithm are all combined the neighborhood information of the gray-scale intensity of the

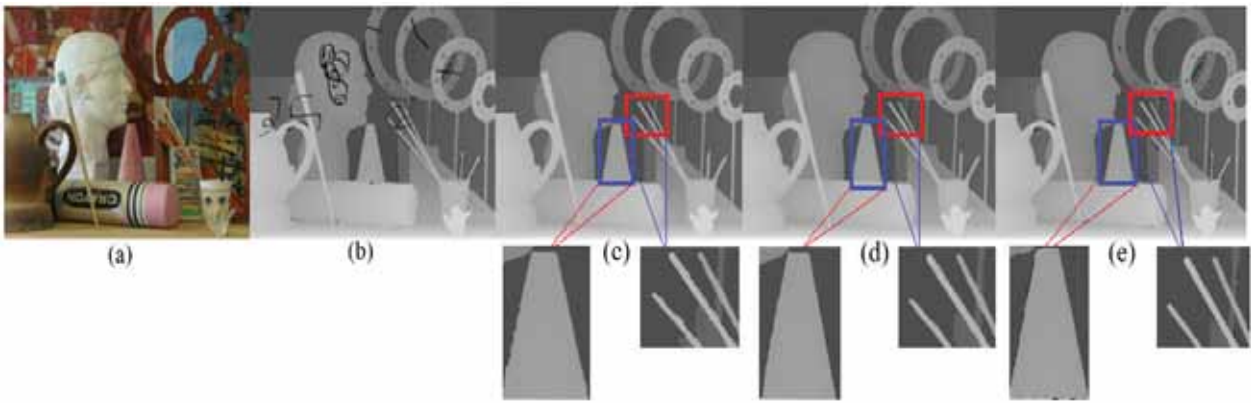

Fig. 6. Restored depth results on the case of *Art* from the Middlebury dataset. (a) RGB image. (b) Degraded depth map. (c) Restored depth image by self-taught regression method [13]. (d) Restored depth image by the proposed method. (e) Ground truth.

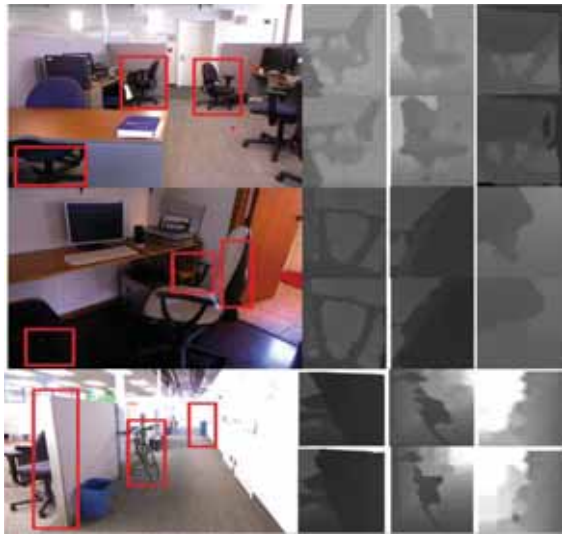

Fig. 5. Details of image depth recovery. The first row on the right of each image is the result of our algorithms depth recovery, the second row is the result of self-taught regression algorithm [13].

RGB image and the corresponding depth map captured by Kinect to predict missing depth values, and learn in a self-taught way. The difference is that our method employs the cooperative profit random forests (CPRF) to predict the depth value. CPRF uses the split criterion of the Banzhaf power index that can be learned and considered the internal relationships between feature variables, i.e., a group of feature variables with strong discrimination ability in term of the target class. In fact, the RGB image and its aligned depth map captured by Kinect representing the same scene, have strong correlations. CPRF can explore this strong correlation and obtain more accurate prediction results.

For the self-taught regression algorithm [13] employs the original random regression forests algorithm that uses the least square error to split the tree node. This split method tends to choose a single feature with a strong discrimination, which pays less attention to the intrinsic structure of the attribute variables and fails to find attributes with a good discriminate ability as a group. Moreover, the self-taught regression algorithm [13] first employs the Make 3D method to obtain a rough depth prediction image. The depth values of the obtained rough depth image are usually inaccurate, because the training data of Make 3D consists of the indoor scenes that do not include the office scenes. The inaccurate depth values bring a negative impact on the prediction of random regression forests. This inevitably leads to inaccurate predictions. In particular, the application scenarios of the self-taught regression algorithm [13] also have limitations, because it is totally dependent on the acquisition of the rough depth.

B. The Middlebury dataset

To further verify the effectiveness of the proposed method, we compared the performance of our algorithm and the method of Yang *et al.* [13] on the Middlebury benchmark dataset [21] with synthesized holes. Moreover, we used the Root Mean Square Error (RMSE) quantitative measure to quantify the performance of the algorithm.

$$\text{Root Mean Square Error:}\sqrt{\frac{1}{N}\sum_{x\in P}(d(x)-\hat{d}(x))^2}$$

where $d(x)$ and $\hat{d}(x)$ are the ground truth depth and restored depth at the pixel x; P is the whole set of pixels in an depth map, and N is the number of the pixels in P.

TABLE I
QUANTITATIVE RECOVERY RESULTS WITH KINECT-LIKE DEGRADATIONS.

Dataset	Yang *et al.* [13]	Ours
Art	4.04	**3.85**

The experimental results are shown in Fig. 6, and the quantitative results in terms of Root Mean Square Error (RMSE) are shown in Table I. From the Table I, we can see

that our method gets the lowest RMSE in the case of *Art*. Fig. 6 and Table I further demonstrate the validity of the proposed method.

V. Conclusion

In this paper, we propose a new method to restore the missing depth data obtained by Kinect. The cooperative profit random forests (CPRF) is trained by using the neighborhood information of the RGB image and the corresponding depth map, and then the trained CPRF is used to predict the missing depth value. The proposed depth recovery method uses a similar to self-taught learning to keep learning and predicting. Moreover, CPRF exploring the strong correlation of features (pixels) from the RGB image and the corresponding depth image can obtain satisfactory results. The experimental results show the effectiveness of the proposed method that can well restore the missing depth values.

Acknowledgement

This work was supported by the EPSRC through project 4D Facial Sensing and Modelling (EP/N025849/1), the Royal Academy of Engineering through the project Multimodal Data-based Mental Workload and Stress Assessment for Assistive Brain-Computer Interface (NRCP1516/1/74), the National Natural Science Foundation Of China (NSFC) (No.61271405) and the International Science & Technology Cooperation Program of China (ISTCP) (No. 2014DFA10410).

References

[1] Z. Xu, X. Qiu, and J. He, "A novel multimedia human-computer interaction (hci) system based on kinect and depth image understanding," in *International Conference on Inventive Computation Technologies*, 2017, pp. 1–6.

[2] G. Choe, J. Park, Y.-W. Tai, and I. So Kweon, "Exploiting shading cues in kinect ir images for geometry refinement," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 3922–3929.

[3] W. Liu, A. S. Mian, A. Krishna, and B. Y. L. Li, "Using kinect for face recognition under varying poses, expressions, illumination and disguise," in *IEEE Workshop on Applications of Computer Vision*, 2013, pp. 186–192.

[4] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE Multimedia*, vol. 19, no. 2, pp. 4–10, 2012.

[5] J. Sun, G. Zhong, J. Dong, H. Saeeda, and Q. Zhang, "Cooperative profit random forests with application in ocean front recognition," *IEEE Access*, vol. 5, no. 99, pp. 1398–1408, 2017.

[6] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 1817–1824.

[7] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664–672, 2004.

[8] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013.

[9] D. Miao, J. Fu, Y. Lu, and S. Li, "Texture-assisted kinect depth inpainting," in *IEEE International Symposium on Circuits and Systems*, 2012, pp. 604–607.

[10] H. Yao, Y. Chen, and C. Ge, "Image inpainting strategy for kinect depth maps," *Proceedings of SPIE*, vol. 8878, 2013.

[11] S. Liu, Y. Wang, H. Wang, and C. Pan, "Kinect depth inpainting via graph laplacian with tv21 regularization," in *Pattern Recognition*, 2014, pp. 251–255.

[12] X. Zhang and X. Wu, "Image interpolation by adaptive 2-d autoregressive modeling and soft-decision estimation," *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, vol. 17, no. 6, p. 887, 2008.

[13] P. Yang, H. Zhao, L. Qi, and G. Zhong, "Self-taught recovery of depth data," in *Asia-Pacific Signal and Information Processing Association Summit and Conference*, 2015, pp. 1270–1275.

[14] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[15] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Learning 3d scene structure from a single still image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 824–840, 2009.

[16] J. Liu, X. Gong, and J. Liu, "Guided inpainting and filtering for kinect depth maps," in *International Conference on Pattern Recognition*, 2012, pp. 2055–2058.

[17] J. Yang, X. Ye, K. Li, C. Hou, and Y. Wang, "Color-guided depth recovery from rgb-d data using an adaptive autoregressive model," *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3443–3458, 2014.

[18] K. Markov and T. Matsui, "Nonnegative matrix factorization based self-taught learning with application to music genre classification," in *IEEE International Workshop on Machine Learning for Signal Processing*, 2012, pp. 1–5.

[19] R. Kemker and C. Kanan, "Self-taught feature learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. PP, no. 99, pp. 1–13, 2017.

[20] K. Zhou, Y. Liu, J. Song, L. Yan, F. Zou, and F. Shen, "Deep self-taught hashing for image retrieval," pp. 1215–1218, 2015.

[21] M. Datasets, "http://vision.middlebury.edu/stereo/data," 2013.