CLASSIFICATION OF MOBILE LIDAR DATA USING VOX-NET AND AUXILIARY TRAINING SAMPLES

Hanxian He*, Kourosh Khoshelham, Clive Fraser

Geomatics Group, Dept. of Infrastructure Engineering, University of Melbourne, Australia- hanxianh@student.unimelb.edu.au, k.khoshelham@unimelb.edu.au, c.fraser@unimelb.edu.au

Commission VI, WG VI/4

KEY WORDS: Vox-Net, SAMME, Object Recognition, Point Cloud, 3DCNN, Deep Learning, Transfer Learning

ABSTRACT:

The classification of mobile Lidar data is challenged by the complexity of objects in the point clouds and the limited number of available training samples. Incomplete shape, noise points and uneven point density make the extraction of features from point clouds relatively arduous. Additionally, the difference in point density, and size and shape of objects, restricts the utilization of labelled samples from other sources. To solve this problem, we explore the possibility of improving the classification performance of a state-of-the-art deep learning method, Vox-Net, by using auxiliary training samples from a different dataset. We compare the performance of Vox-Net trained with and without the auxiliary dataset. The comparison shows that more instances can be recognized in classes with auxiliary data. At the same time, the performance in classes without complementary data can deteriorate due to the low number of samples in these categories. To achieve a balance in the performance for different categories, we further replace the classification layer of Vox-Net with AdaBoost. The AdaBoost classification displays good recognition ability in classes with few instances but decreases the overall accuracy.

1. INTRODUCTION

A common challenge in segment-based classification of Lidar point clouds is the generation of training samples. On the one hand, it is difficult to obtain enough balanced training samples for training by labelling the segments manually. On the other hand, samples from publicly available datasets exhibit a wide range of data formats, precision and structure, which makes it difficult to utilize these datasets for training.

Traditional feature-based classification methods require a considerable number of training samples to achieve a satisfactory performance(He et al., 2017). Recent feature learning methods based on deep learning need an even larger set of training samples. Fehr et al. (2016), Jing and Suya (2015), and Yokoyama et al. (2013) achieved successful pole-like object detection, but less than good performance in the classification of pole-like objects due to the limitation of training samples. The diversity of objects in mobile Lidar point clouds makes it even more difficult to balance the number of training samples. One approach to reducing the number of required training samples is to reduce the dimensionality by feature selection methods (Khoshelham et al., 2013).

In order to effectively reduce the dimension of features, Chatfield et al. (2011) employed feature encoding methods to preserve the discriminative power of the features. Recent research on 3D deep convolutional networks shows that better performance can be achieved by feature learning rather than feature selection or encoding. However, the deep learning method is inherently a data-hungry method (Koch et al., 2015). To reduce sample number requirements, one-shot learning (Koch et al., 2015) and transfer learning (Dai et al., 2007; Rosenstein et al., 2005) have been introduced for object classification.

A feasible way to enrich the training data for point cloud classification is to incorporate samples from other Lidar datasets, or from Web-based libraries of 3D models, as auxiliary training samples. The shape information can be transferred within multiple data representations and formats (Kar et al., 2015; Qi et al., 2016; Su et al., 2015). Moreover, the data collected and labelled before can be reused in a new task by transfer learning.

Considering that the objects obtained from different Lidar datasets vary in resolution, density and even formats, we adopted Vox-Net (Maturana and Scherer, 2015b), which can adapt to different sizes and density of input data. Maturana and Scherer (2015b) demonstrate that Vox-Net can handle different datasets, such as Lidar data, CAD data and RGB-D data, in the same network framework by adjusting the grid algorithm and resolution. Compared with other 3D deep learning nets, Vox-Net can outperform other nets in low-quality mobile lidar dataset.

In this paper, we design several experiments to test the performance of classification with and without an auxiliary data. Data with different scales and resolution are also introduced to the check the scale and resolution invariance of volumetric grid and Vox-Net based framework. Moreover, AdaBoost classification is introduced to evaluate the performance of weight adjustment method in instance-based transfer learning.

* Corresponding author

https://doi.org/10.5194/isprs-archives-XLII-2-W13-1001-2019 | © Authors 2019. CC BY 4.0 License.

2. RELATED WORK

Although deep learning methods show outstanding performance in classification of 2D images, their extension to the 3D domain remains severely challenging. First of all, effective representation and organization of 3D point clouds are required for convolutional neural network (CNN) filters. Secondly, the number of point clouds collected could vary in different regions of the object, which could influence the organization of point clouds for CNN filters. Different data representation methods are proposed to deal with these two issues. In PointNet (Charles et al., 2017) and PointNet++ (Fan et al., 2017), the network learns from the original points, and optimization functions are designed for selecting informative points. Another popular solution is the volumetric representation, in which the raw point clouds are voxelized into uniform grids (Maturana and Scherer, 2015a; Minto et al., 2018; Qi et al., 2016; Wu et al., 2015; Zhou and Tuzel, 2017). Although the voxelization of point clouds into regular 3D voxel grids makes it easier for weight sharing and kernel optimizations, this quantization can introduce unnecessary variance to the original data, and the performance of 3D CNNs varies according to the volume resolution. Another attempt at adopting deep learning on point clouds is to construct graphs of points in Euclidean space. The Kd-network (Klokov and Lempitsky, 2017) and octree-based network (Wang et al., 2017) have been proposed for the organization of points. 2.5D convolutional neural networks solve the issue by reducing the dimension into 2D with multiple views (Qi et al., 2016) or depth images (Roveri et al., 2018). In addition, 3D deep feature-based networks (Wang et al., 2018) and geometric 3D nets (Weinmann et al., 2017) have also been proposed.

Besides the challenges in extending 2D deep learning models into 3D, the number of training samples is another factor influencing the performance of 3D deep learning-based classification. Incorporating data or basic networks from other domains is popular in 2D classification (Weiss et al., 2016). Generally, the situation in which previously learned knowledge from other domains, tasks or distributions is reused in a current machine learning task is named transfer learning (Pan and Yang, 2010). Tan et al. (2018) provide a comprehensive review of available deep learning transfer learning methods. Instance-based transfer learning is widely used in similarly structured data by adjusting the sample selection bias (Dai et al., 2007; Yao and Doretto, 2010) to pick out the partial useful instances in the training section. In mapping-based transfer learning, instances from the source domain and target domain are transferred into a new domain sharing more similarity. Transfer components analysis (TCA) and joint maximum mean discrepancy (JMMD) provide possible mapping solutions (Long et al., 2017; Zhang et al., 2017). Network-based transfer learning is widely used in deep learning networks by reusing the pre-trained front-layers (Oquab et al., 2014). Adversarial-based transfer learning is proposed by Goodfellow et al. (2014) to find transferable representations for both domains.

Here, in order to incorporate datasets collected with different equipment in different scenarios, we design instance-based transfer learning experiments based on Vox-Net and AdaBoost.

3. METHODOLOGY

The conceptual framework for the proposed method is shown in Figure 1. It includes data processing, Vox-Net training and classification. In the data processing step, the segments are first augmented into 12 rotated copies, then voxelized into grid format from the point clouds. In the second step, we train the Vox-Net on an original dataset with and without an auxiliary dataset. The performance of Vox-Net-based classifications with and without the complementary dataset is then compared. Finally, AdaBoost classification is implemented on the features obtained from the output layer of Vox-Net trained with the combined dataset.

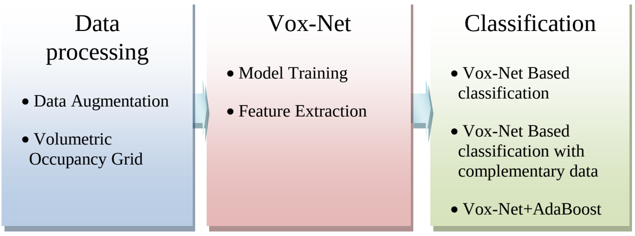
Data processing
• Data Augmentation
• Volumetric Occupancy Grid
Vox-Net
• Model Training
• Feature Extraction
Classification
• Vox-Net Based classification
• Vox-Net Based classification with complementary data
• Vox-Net+AdaBoost

Figure 1. The conceptual framework of the Vox-Net based feature extraction and classification.

3.1 Data Processing

We first unite the two data sources into point segments after segmentation of original points. Then we select perfect training samples from the auxiliary dataset. Then we follow the basic workflow of Vox-Net after augmentation of training samples. In the grid section, the resolution in the gridding of point clouds is set as 0.2m considering the length of light poles in the complementary dataset. We compare the performance under distinct settings of rotation interval and augmentation number, and then get the 12 rotated copies with a fixed interval value along the z coordinate. Volumetric occupancy grid is realized by the hit grid method, which calculates the point occupancy information in each cell of the grid. As no origin or orientation information is provided, the mean value of samples is used as the origin and the samples are assumed perpendicular to the ground.

3.1.1 Data Augmentation

It is obvious that the origin, orientation, and the resolution of the voxel grid can influence the representation of gridded segments. In the grid occupancy part, the centroid of segment is calculated by $\left[mean(x_j), \mathrm{m}ean(y_j), \mathrm{m}ean(z_j) \right]$, and the rotation interval is set as $2\pi/(12+1)$ along z coordinate. Then, the coordinates of the points are normalized. Based on the shape size of most road furniture considered in this research, the size and resolution of a voxel are set as $\mathbf{p} = (32,32,32)$ and $r = 0.2m$ respectively. The basic geometric information, including the majority part of the segments in the voxel box, is thus maintained. Rotation augmentation, together with translation before the gridding, is employed. Random augmentation of the dataset is adopted by creating $n=12$ copies along the Z axis, with equal interval rotation within $(-\pi,\pi)$ and normalized translation. (Sedaghat et al., 2016).

3.1.2 Volumetric Occupancy Grid

The instances from two data source are different in resolution, scale and quality. The instances from two data sources are united into identical data structure by utilizing grid and Vox-Net. 3D ray tracing (Amanatides and Woo, 1987), which builds the occupancy model by calculating the number of hits or passthroughs for each voxel, is the most popular way to realize gridding in range data. Here, we selected the hit grid model in our method. In this grid model, cell occupied with points is set as 1, otherwise 0.

3.2 Vox-Net Modelling

The framework of Vox-Net is shown in Figure 2. Like most 3D CNNs, it consists of an input layer, convolutional layers, pooling layer and a fully connected layer. In the input layer, the volumetric grid of fixed size $32*32*32$ voxels is accepted. The information in the grid is the occupancy information. It can also be updated with other features (Zhou and Tuzel, 2017). We set the learning rate at 0.001, batch size at 32, number of batches in one epoch at 5000 and number of epochs at 8. For the adaptation function we used AdamOpitimizer. The number of other parameters can be referenced in Figure 2. In the training section, the pooling layer is introduced to decrease the chance of overfitting and achieving translation invariance in the deep learning net.

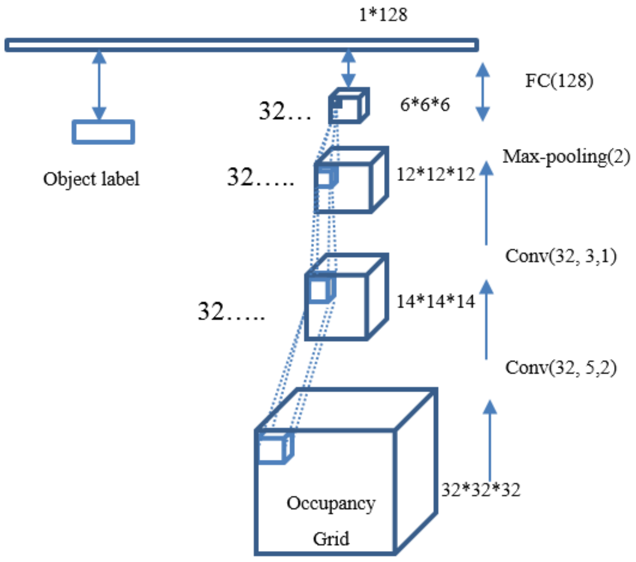

Figure 2. The conceptual framework of Vox-Net based feature extraction and classification.

3.3 Classification

In a first experiment, we use Vox-Net with the SoftMax function to realize classification, and to compare the results of models trained with both the original dataset only and the combined dataset. In a second experiment, we first extracted the feature vectors of samples in the fully connected layer of the Vox-Net model trained with the combined dataset. AdaBoost classification on these generated feature vectors was then implemented. We select AdaBoost (Freund and Schapire, 1996) in view of its principle of improving the classification by increasing the weight of misclassified instances. SAMME (Hastie et al., 2009) was selected as the basic AdaBoost algorithm.

4. EXPERIMENT AND ANALYSIS

4.1 Data Description

The Sydney dataset contains a variety of common urban road objects scanned with a Velodyne HDL-64E Lidar, the data having been collected by De Deuge et al. (2013) in the CBD of Sydney, Australia. The Sydney dataset contains 631 urban objects in segmented format with XYZ and range information. In the experiment, we selected the same target classes and organization of the datasets adopted in Vox-Net for comparison purpose. A set of 588 objects was selected from the full 631, split evenly into 4 folds. Folds were selected to ensure no object appeared in two folds (some objects appear more than once in the dataset, scanned from different positions).

The Enschede dataset, which covers several similar urban road object categories as the Sydney dataset, is used as the auxiliary training dataset in the experiment. The Enschede dataset was collected with an Optech LYNX Mobile Mapper system by the German company TopScan in 2008 in the city of Enschede, Netherlands.

The number of samples in each category in each file is provided in Table 1. F0, F1, F2 and F3 are the datasets collected in Sydney. The first three folds from F0 to F2 are set as the original training datasets, and F3 is reserved as the testing dataset. The samples in these four folds are evenly distributed and the accuracy did not show significant differences when the test folder was set differently. F4 is the dataset collected in Enschede, which is used as a complementary training dataset.

Table 1. The distribution of samples.

Category	Dataset 1 (Sydney Dataset) Number				Dataset 2 (Enschede Dataset) Number
File Name	F0	F1	F2	F3	F4
4wd	5	6	4	6	
building	5	5	5	5	
bus	5	3	3	5	
car	23	20	21	24	34
pedestrian	37	36	34	45	19
pillar	6	5	4	5	
pole	6	5	4	6	
traffic light	10	18	8	11	25
traffic sign	11	18	11	11	33
tree	8	8	8	10	92
truck	3	3	3	3	
trunk	14	13	15	13	
ute	4	4	4	4	
van	9	11	8	7	
Total Number	146	155	132	155	203

4.2 Experiments and Results

In the first experiment, we compared the performance of Vox-Net based classification with and without the Enschede dataset. In the training section, each instance of the training datasets is augmented with 12 copies. In the test section, the original samples were used to evaluate the performance. The recalls, precision and F1-score in each category found for Vox-Net based classification are provided in Figure 3, Figure 4 and Figure 5. The SoftMax function is used here for classification.

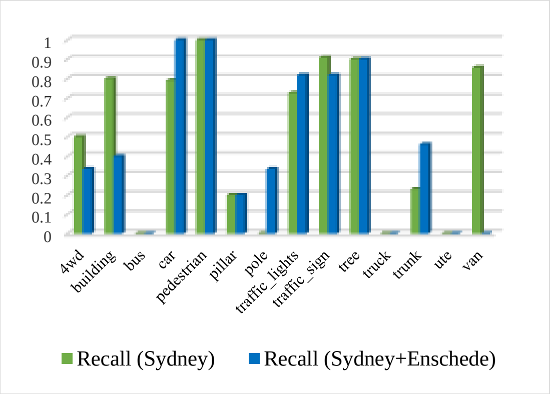

Figure 3. The Recall of Vox-Net classification with and without a complementary dataset.

The trained Vox-Net model in both cases achieved similar evaluation and test accuracy. The overall accuracy of the model trained with the Sydney dataset is 69.68%. The accuracy of the model trained with the combined dataset are 70.32%. The number of correctly recognized objects is provided in Table 2. Classification accuracy was limited by the quality of Sydney dataset, with misclassification mostly occurring within the vehicle- and pole-structure objects.

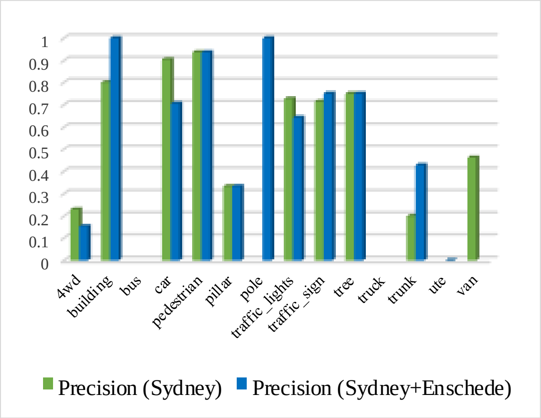

Figure 4. The Precision of Vox-Net classification with and without a complementary dataset.

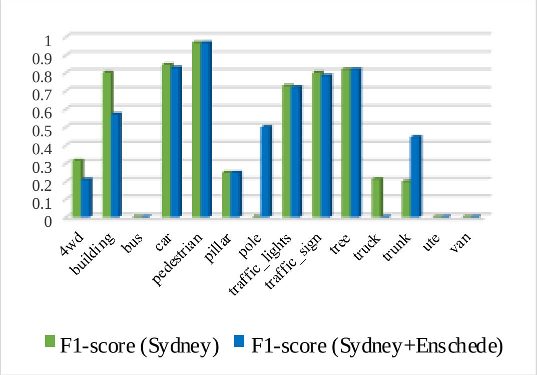

Figure 5. The F1-score of Vox-Net classification with and without a complementary dataset.

From the comparison, it was found that (1) the classification results in pedestrian and tree categories, which have enough training samples, remained the same; (2) the misclassification between pole and tree trunks was mitigated by the increase of instances from several pole-structure categories by checking the confusion matrixes; and (3) the increase of instances from class car only in vehicle-type objects improved the number of correctly recognized instances in the car class, while decreasing the possible recognition of other vehicle-type objects.

In the second experiment, AdaBoost was combined with Vox-Net to improve the classification result. First, we trained the Vox-Net model with the Sydney and Enschede datasets together. Then, AdaBoost classification was implemented on the feature vectors extracted in the fully connected layer of the trained Vox-Net model. Multi-class AdaBoost decision trees proposed by Hastie et al. (2009) were used. The number of estimators was set at 200, and the learning rate at 0.9. The parameter and algorithm are selected according to optimization tests. The recall, precision and F1-score with the SoftMax function and SAMME AdaBoost classification are provided in Figure 6 Figure 7 and Figure 8 separately.

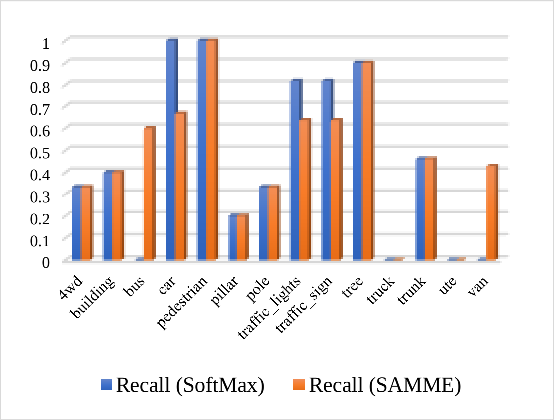

Figure 6. The Recall of SoftMax and SAMME algorithms on feature instances extracted from the fully connected layer of the trained Vox-Net model.

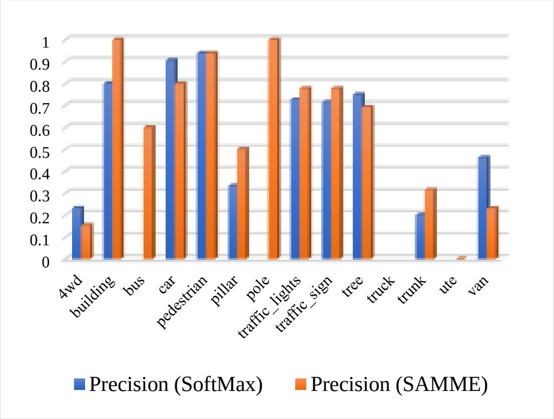

Figure 7. The Precision of SoftMax and SAMME algorithms on feature instances extracted from the fully connected layer of the trained Vox-Net model.

https://doi.org/10.5194/isprs-archives-XLII-2-W13-1001-2019 | © Authors 2019. CC BY 4.0 License.

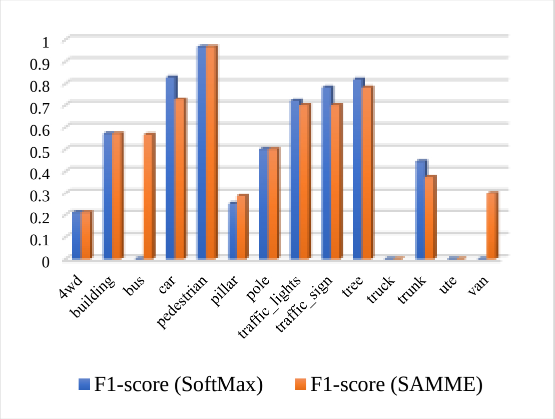

Figure 8. The F1-score of SoftMax and SAMME algorithms on feature instances extracted from the fully connected layer of the trained Vox-Net model.

The result from AdaBoost showed higher recall and precision in most classes with limited instances. Also, it alleviated the misclassification within vehicle-type objects. We also observed that classes with few instances, say bus, truck and ute with instance less than 12, cannot be correctly recognized in any of the cases. However, the AdaBoost method did not improve the overall accuracy compared with the SoftMax function in the Vox-Net model.

Upon further investigation of the results listed in Table 2, we found that the incompleteness of vehicle shape and the variation of traffic signs slightly influences the classification result. Additionally, the size of both buses and buildings, which were larger than 32*0.2m, made the distinguishing of these two categories difficult. In the experiments, the scale of light poles in the Enschede dataset was different to that in the Sydney dataset. The framework also showed scale-invariance in classification of light-pole by introducing multi-scale light poles.

Table 2. The comparison of correctly recognized objects in each class from three experiments. D1 represents the Sydney dataset, and D2 represents the Enschede dataset.

Label	Real Number	Recognized Number		
		VoxNet +Sydney Dataset	VoxNet+ Sydney +Enschede Dataset	SAMME+ VoxNet
4wd	6	3	2	2
building	5	4	2	2
bus	5	0	0	3
car	24	19	24	16
pedestrian	45	45	45	45
pillar	5	1	1	1
pole	6	0	2	2
traffic light	11	8	9	7
traffic sign	11	10	9	7
tree	10	9	9	9
truck	3	0	0	0
trunk	13	3	6	6
ute	4	0	0	0
van	7	6	0	3
Total	155	108	109	103

5. DISCUSSION

The performance of Vox-Net based mobile lidar classification with auxiliary data has been investigated, and it has shown that the occupancy grid and Vox-Net based framework can united the original and complementary data in an efficient way.

To better demonstrate the effectiveness of this transfer learning framework, we will aim to improve the following aspects: (1) set comparison experiments by setting the instance number of the original datasets at several levels from sparse to medium to sufficient; (2) set comparison experiments by changing the instance ratio of the datasets in the target domain to the source domain; (3) improve the weight adjustment algorithm to filter out dissimilar instances from the source domain and improve the recognition of instances misclassified in the target domain at the same time; and (4) data in other formats will be introduced to evaluate the proposed method.

REFERENCES

Amanatides, J., Woo, A., 1987. A fast voxel traversal algorithm for ray tracing, Eurographics, pp. 3-10.

Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation, Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. IEEE, pp. 77-85.

Chatfield, K., Lempitsky, V.S., Vedaldi, A., Zisserman, A., 2011. The devil is in the details: an evaluation of recent feature encoding methods, BMVC, p. 8.

Dai, W., Yang, Q., Xue, G.-R., Yu, Y., 2007. Boosting for transfer learning, Proceedings of the 24th international conference on Machine learning. ACM, pp. 193-200.

De Deuge, M., Quadros, A., Hung, C., Douillard, B., 2013. Unsupervised feature learning for classification of outdoor 3d scans, Australasian Conference on Robitics and Automation, p. 1.

Fan, H., Su, H., Guibas, L.J., 2017. A point set generation network for 3d object reconstruction from a single image, Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 605-613.

Fehr, D., Beksi, W.J., Zermas, D., Papanikolopoulos, N., 2016. Covariance based point cloud descriptors for object detection and recognition. Computer Vision and Image Understanding 142, 80-93.

Freund, Y., Schapire, R.E., 1996. Experiments with a new boosting algorithm, Icml, pp. 148-156.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets, Advances in neural information processing systems, pp. 2672-2680.

Hastie, T., Rosset, S., Zhu, J., Zou, H., 2009. Multi-class adaboost. Statistics and its Interface 2, 349-360.

He, H., Khoshelham, K., Fraser, C., 2017. A two-step classification approach to distinguishing similar objects in mobile LiDAR point clouds. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences 4.

Jing, H., Suya, Y., 2015. Pole-like object detection and classification from urban point clouds, 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 3032-3038.

Kar, A., Tulsiani, S., Carreira, J., Malik, J., 2015. Category-specific object reconstruction from a single image, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1966-1974.

Khoshelham, K., Oude Elberink, S.J., Xu, S., 2013. Segment-based classification of damaged building roofs in aerial laser scanning data. IEEE Geoscience and Remote Sensing Letters 10, 1258-1262.

Klokov, R., Lempitsky, V., 2017. Escape from cells: Deep kd-networks for the recognition of 3d point cloud models, Computer Vision (ICCV), 2017 IEEE International Conference on. IEEE, pp. 863-872.

Koch, G., Zemel, R., Salakhutdinov, R., 2015. Siamese neural networks for one-shot image recognition, ICML Deep Learning Workshop.

Long, M., Zhu, H., Wang, J., Jordan, M.I., 2017. Deep transfer learning with joint adaptation networks, Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, pp. 2208-2217.

Maturana, D., Scherer, S., 2015a. VoxNet: A 3D Convolutional Neural Network for real-time object recognition, 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 922-928.

Maturana, D., Scherer, S., 2015b. VoxNet: A 3D Convolutional Neural Network for real-time object recognition.

Minto, L., Zanuttigh, P., Pagnutti, G., 2018. Deep Learning for 3D Shape Classification based on Volumetric Density and Surface Approximation Clues.

Oquab, M., Bottou, L., Laptev, I., Sivic, J., 2014. Learning and transferring mid-level image representations using convolutional neural networks, Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1717-1724.

Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Transactions on knowledge and data engineering 22, 1345-1359.

Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J., 2016. Volumetric and multi-view cnns for object classification on 3d data, Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5648-5656.

Rosenstein, M.T., Marx, Z., Kaelbling, L.P., Dietterich, T.G., 2005. To transfer or not to transfer, NIPS 2005 Workshop on Transfer Learning.

Roveri, R., Rahmann, L., Oztireli, C., Gross, M., 2018. A network architecture for point cloud classification via automatic depth images generation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4176-4184.

Sedaghat, N., Zolfaghari, M., Amiri, E., Brox, T., 2016. Orientation-boosted voxel nets for 3D object recognition. arXiv preprint arXiv:1604.03351.

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3d shape recognition, Proceedings of the IEEE international conference on computer vision, pp. 945-953.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C., 2018. A Survey on Deep Transfer Learning. arXiv preprint arXiv:1808.01974.

Wang, P.-S., Liu, Y., Guo, Y.-X., Sun, C.-Y., Tong, X., 2017. O-cnn: Octree-based convolutional neural networks for 3d shape analysis. ACM Transactions on Graphics (TOG) 36, 72.

Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2018. Dynamic graph CNN for learning on point clouds. arXiv preprint arXiv:1801.07829.

Weinmann, M., Jutzi, B., Mallet, C., 2017. Geometric features and their relevance for 3d point cloud classification. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 4, 157.

Weiss, K., Khoshgoftaar, T.M., Wang, D., 2016. A survey of transfer learning. Journal of Big Data 3, 9.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J., 2015. 3D ShapeNets: A deep representation for volumetric shape modeling, CVPR, p. 3.

Yao, Y., Doretto, G., 2010. Boosting for transfer learning with multiple sources, Computer vision and pattern recognition (CVPR), 2010 IEEE conference on. IEEE, pp. 1855-1862.

Yokoyama, H., Date, H., Kanai, S., Takeda, H., 2013. Detection and classification of pole-like objects from mobile laser scanning data of urban environments. International Journal of CAD/CAM 13, 1-10.

Zhang, J., Li, W., Ogunbona, P., 2017. Joint geometrical and statistical alignment for visual domain adaptation. arXiv preprint arXiv:1705.05498.

Zhou, Y., Tuzel, O., 2017. Voxelnet: End-to-end learning for point cloud based 3d object detection. arXiv preprint arXiv:1711.06396.