

AIRBORNE LIDAR POINT CLOUD CLASSIFICATION FUSION WITH DIM POINT CLOUD

Mengdie Zhou, Zhizhong Kang *, Zhen Wang, Min Kong

Dept. of Land Science and Technology, China University of Geosciences Beijing, Haidian District, Beijing, China –
mengdie_zhou@126.com, zzkang@cugb.edu.cn, comige@gmail.com, minkong0813@163.com

Commission II, WG II/3

KEY WORDS: Airborne LiDAR, Dense Image Matching, Point Cloud, Classification, Multi-class TrAdaboost Algorithm, Super-voxel

ABSTRACT:

Airborne Light Detection And Ranging (LiDAR) point clouds and images data fusion have been widely studied. However, with recent developments in photogrammetric technology, images can now provide dense image matching (DIM) point clouds. To make use of such DIM points, a sample selection framework is introduced. That is, first, the geometric features of LiDAR points and DIM points are extracted. Each feature per point is considered a sample. Then we extend the binary TrAdaboost classifier into a multi-class one to train all the samples. The classifier automatically assigns weights to the samples in the DIM points. The useful samples are assigned large weights and consequently impact the classification results largely and vice versa. As a result, the useful samples of the DIM points are kept to improve on the LiDAR points classification performance. Because only the samples are used, the registration between the DIM points and LiDAR points is not needed. Moreover, the DIM points capturing similar classes but not the same scene as the LiDAR points can also be used. By our framework, existing aerial images can be fully utilized. For testing the generation ability, the framework is applied in a super-voxel-based classification approach by replacing the points-based features with the super-voxel-based features. In the experiments, whether DIM points at the same places as those of LiDAR are used or not, the results after fusion show that, the LiDAR points classification performance has improved. Also, the better the quality of DIM points are, the better the classification performance achieves.

1. INTRODUCTION

Airborne LiDAR can directly obtain high-precision, high-density 3D coordinates, which is widely used for applications like 3D reconstruction and scene understanding (Sohn et al., 2008; Sampath et al., 2010), in which point cloud classification plays a critical role.

Many methods have been proposed for point cloud classification, ranging from point-based classification to object-based classification. For point-based classification, Sun et al. (2014) extracted geometric features from point clouds and classified them using a random forest classifier (Fan et al., 2013). Niemeyer et al. (2013) integrated a Random Forest (RF) classifier and a Conditional Random Field (CRF) (Niemeyer et al., 2012) for multi-class classification. Due to the high level of noise in point clouds, the point-based classification results often have a "pepper and salt" phenomenon. Therefore, many studies clustered the point clouds into objects. Kang et al. (2018a) voxelized a scene of point clouds and extract features from the voxels to recognize pole-like objects. Huang et al. (2016) turned the point clouds into 3D voxels and use a 3D Convolutional Neural Network (CNN) for classification. Ramiya et al. (2016) used a super-voxel for point clouds classification and showed that the super-voxel can improve the computational efficiency of dense point clouds. Ahmad et al. (2014) segmented point cloud into super-voxel by a link-chain method and used geometrical models and local descriptors of the super-voxel for classification. Zhang et al. (2016) divided the point cloud into hierarchical point clusters and used the sparse coding model (Yang et al., 2009) and latent Dirichlet allocation model (Blei et

al., 2003) to extract and encode the shape features of the hierarchical point clusters for classification.

To further improve classification performance, the fusion of optical images and point clouds has been given wide attention. Haala et al. (1999) combined multispectral images and laser altimeter data in an integrated classification for the extraction of buildings, trees, and grass-covered areas. Cao et al. (2012) fused point clouds with its co-registered images (i.e. aerial color images containing red, green and blue (RGB) bands and near-infrared (NIR) images) and other derived features for accurate urban land-cover classification. Guo et al. (2011) proposed a multi-source framework by combining multi-echo LiDAR data, full-waveform LiDAR data and multispectral image data to classify dense urban scenes.

Although a lot of research has been done on the classification of point cloud by fusing images and points, there is still room for improvement. Most of these studies focus on the spectral information in the images. However, currently, a large number of airborne images or especially the unmanned aerial vehicle (UAV) images can provide DIM point clouds via photogrammetric means (Rosnell et al., 2012). The DIM points have also been used for ground objects classification in some studies (Thiel et al., 2017; He et al., 2018; Zhao et al., 2018). In other words, images can not only provide spectral information, but also spatial information. In order to use this spatial information, in this study, a sample selection framework is presented to classify LiDAR points, by using the DIM points as the auxiliary data. The contributions of the framework are twofold.

* Corresponding author

(1) In the framework, a multi-class TrAdaboost algorithm is introduced to automatically select the samples from the DIM points based on the extracted features to improve the classification performance of the LiDAR points. Two widely used types of classification approaches, which are point-based classification, and super-voxel-based classification are used for testing the generation of the framework. The results show the framework can fuse DIM points with LiDAR points to improve the LiDAR points classification performance.

(2) The framework only selects the features of DIM points which are useful for LiDAR points classification, so registration of the DIM points and LiDAR points is not needed. The images containing the classes in LiDAR point clouds can be used. This decreases the data required during the fusion process as other studies did and can make full use of the existing aerial images.

2. METHOD

The feature-based dense matching method can generate point clouds from airborne images or UAV images. However, the generation of DIM points needs to match the image, and the lack of image texture leads to the lack of DIM points (Feng, 2014). Figure 1 shows an example of DIM points and LiDAR points. It can be seen that the two kinds of point clouds do not look the same. The LiDAR points are more regular and have less data missing, compared to the DIM points. Also, there is a lack of data on the sides of buildings and the bottom of vegetation, as shown in the yellow boxes in Figure 1.

In order to effectively mine the useful information from the DIM points, a sample selection framework is proposed, and the flow chart is as shown in Figure 2. First, the features are individually extracted from the LiDAR points and DIM points. Then they are fed into the TrAdaboost algorithm which selects useful information from the DIM points to classify the LiDAR points.

2.1 Feature Extraction

This study focuses on highlighting the use of the DIM points, so only the geometric features are used and the spectrum of images is not considered here. Thus, first, the support region is delineated. Let \mathbf{q} be a point in the point cloud. Let $\mathbf{N}_q = \{\mathbf{p} \mid \mathbf{p} \text{ is one of the } \mathbf{k} \text{ closest points of } \mathbf{q}\}$ be the support region of point \mathbf{q} and in this study, $\mathbf{k} = 90$. Let us also assume $\bar{\mathbf{p}}$ to be the centroid of all points in \mathbf{N}_q . Then the geometric features for \mathbf{q} used in this paper are introduced as follows.

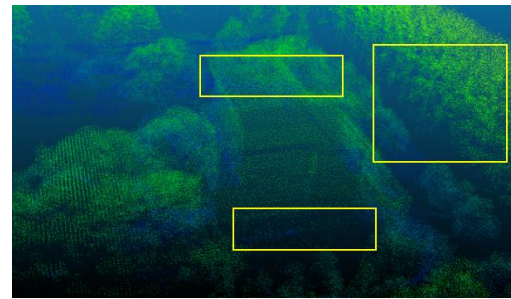
1) Height-based features: The normalized height N_H (Niemeyer et al., 2013) is expressed using the difference between the DSM and DTM, which can eliminate the effect of topographic relief.

2) Eigenvalue-based features: The eigenvalues λ_1, λ_2 , and λ_3 ($\lambda_1 \geq \lambda_2 \geq \lambda_3$) are obtained by finding the covariance (\mathbf{C}_p) of \mathbf{N}_q (Weinmann et al., 2014):

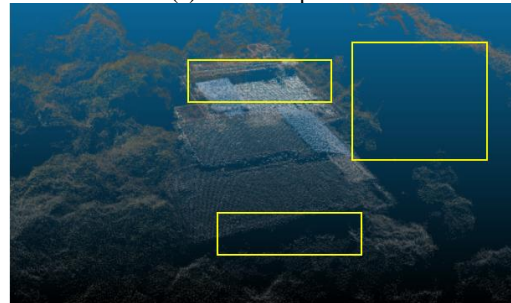
$$\mathbf{C}_p = \frac{1}{|N|} \sum_{\mathbf{p} \in \mathbf{N}} (\mathbf{P}_i - \bar{\mathbf{P}})(\mathbf{P}_i - \bar{\mathbf{P}})^T \quad (1)$$

where \mathbf{T} is the transpose operator

The ranges of the eigenvalues computed for \mathbf{C}_p of different points are different. To compare these eigenvalues of different covariance matrices, the eigenvalues of each covariance matrix



(a) LiDAR points



(b) DIM points

Figure 1. Comparison between LiDAR points and DIM points

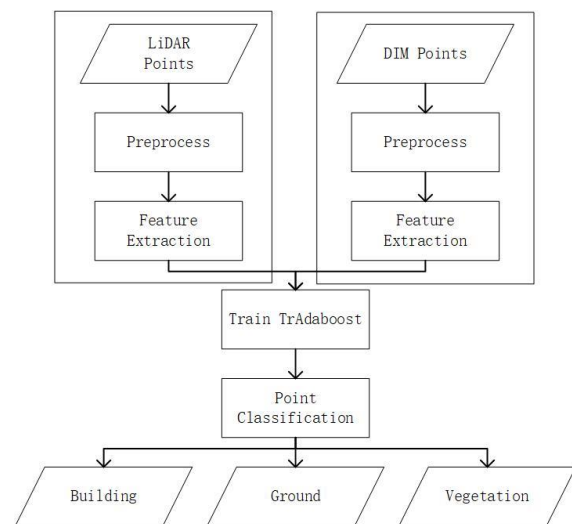


Figure 2. The flow chart of this framework

need to be normalized. That is

$$\lambda_i = \frac{\lambda_i}{\sum_i \lambda_i}, i = 1, 2, 3 \quad (2)$$

Based on solving eigenvalues, we can derive features to distinguish planes, edges, lines and more. That is, anisotropy A_λ , planarity P_λ , sphericity S_λ , and linearity L_λ .

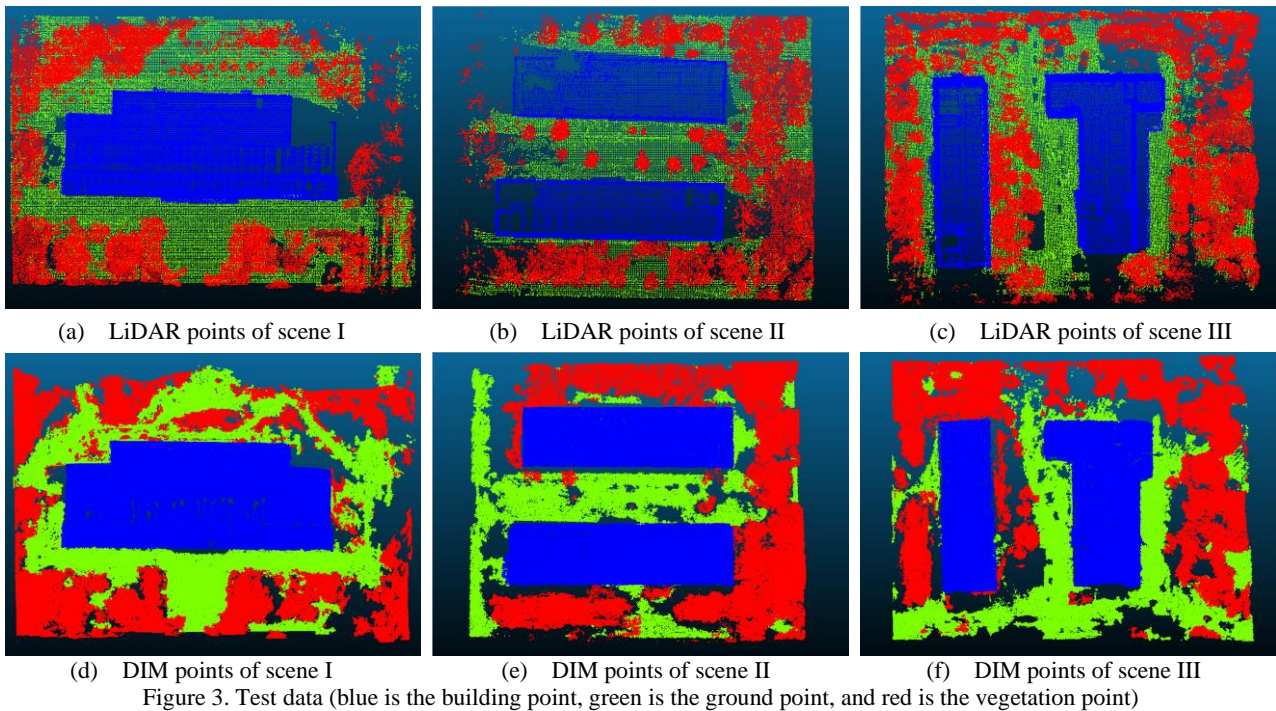
$$A_\lambda = (\lambda_1 - \lambda_3)/\lambda_1 \quad (3)$$

$$P_\lambda = (\lambda_2 - \lambda_3)/\lambda_1 \quad (4)$$

$$S_\lambda = \lambda_3/\lambda_1 \quad (5)$$

$$L_\lambda = (\lambda_1 - \lambda_2)/\lambda_1 \quad (6)$$

3) The normal vector: A plane can be fitted bases on the points in \mathbf{N}_q as Equation (7). Then the normal direction (\mathbf{n}) of



		Building	Ground	Vegetation	Total
Scene I	LiDAR points	127386	43148	91148	261682
	DIM points	125027	104551	120577	350155
Scene II	LiDAR points	44627	24719	66324	135670
	DIM points	122651	46359	85924	254934
Scene III	LiDAR points	56801	41505	101174	199480
	DIM points	131402	90846	138996	361244

Table 1. The number of the two kinds of point clouds of each class in each scene

	Test 1	Test 2	Test 3
Train set	LiDAR points and DIM points in scene I	LiDAR points and DIM points in scene II	LiDAR points and DIM points in scene III
Test set	LiDAR points in scene II	LiDAR points in scene III	LiDAR points in scene I

Table 2. Train set and test set in three tests

the plane is taken as the normal vector of \mathbf{q} , which can be expressed in three directions (n_x, n_y, n_z) as:

$$p(\mathbf{n}, d) = \arg \min_{(\mathbf{n}, d)} \sum_{i=1}^k (\mathbf{n} \mathbf{p}_i - d)^2 \quad (7)$$

where d is the distance to the origin of the coordinates

4) Curvature, Roughness (Zhao et al., 2018): Curvature (cur) reflects the shape and plane characteristics of the point cloud, which can effectively distinguish building points and vegetation points. Roughness (rou) refers to the average value of the distances from all points in the support region to the local geometric plane.

In this paper, the required geometric feature variables are combined into a 13-dimensional feature vector \mathbf{F}_v as Equation (8).

$$\mathbf{F}_v = [N_H, \lambda_1, \lambda_2, \lambda_3, A_\lambda, P_\lambda, S_\lambda, L_\lambda, n_x, n_y, n_z, cur, rou] \quad (8)$$

2.2 The Multi-class TrAdaboost Algorithm

After the \mathbf{F}_v of all the points in the DIM points and the LiDAR points are calculated, a classifier is used to classify the LiDAR points. Here, RF, which has been widely used and shows good performance for classification (Chehata et al., 2009) is used. However, RF cannot integrate the information in the DIM points into the training.

Thus the TrAdaboost algorithm (Dai et al., 2007), which can automatically select useful samples from the DIM points is employed. However, the classical TrAdaboost algorithm is usually only applicable to binary classification problems (Chen et al., 2019). Thus in this paper, the binary TrAdaboost is extended into a multi-class TrAdaboost. The details of the algorithm are shown as follows:

- 1) Initialize the weights (w) for all the samples to 1.

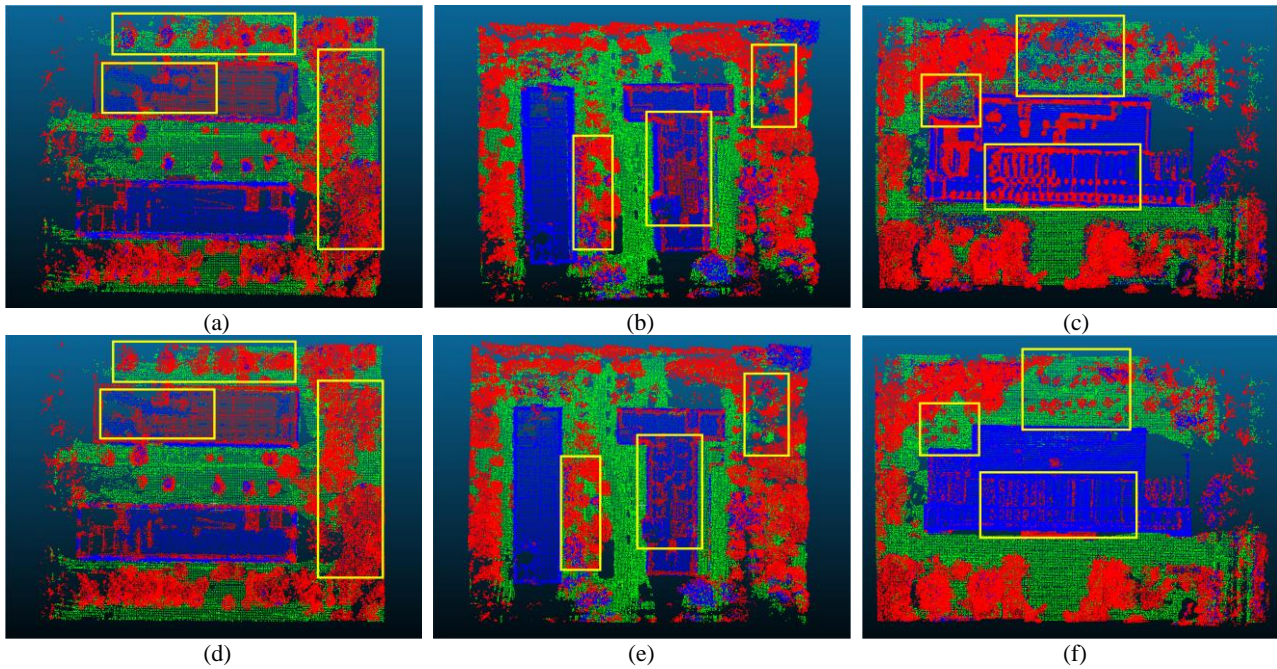


Figure 4. Point-based classification results. (a) - (c) are the results by using LiDAR points for scene II, III and I. (d) - (f) are the results by using LiDAR points and DIM points for scene II, III and I. Buildings, ground, and vegetation are colored in blue, green, and red.

category		Building(%)	ground(%)	vegetation(%)	Accuracy(%)
Test 1, Scene II	LiDAR	57.21/83.20/67.80	97.53/86.41/91.63	91.20/78.47/84.36	81.17
	LiDAR + DIM	52.01/92.29/66.53	97.69/87.58/92.36	96.36/77.05/85.63	82.01
Test 2, Scene III	LiDAR	73.33/78.28/75.72	97.81/94.04/95.89	88.61/86.96/87.78	86.17
	LiDAR + DIM	73.12/81.40/77.04	98.18/92.94/95.48	90.22/87.25/88.71	87.01
Test 3, Scene I	LiDAR	62.26/87.25/72.66	66.75/98.14/79.46	93.36/60.17/73.18	73.83
	LiDAR + DIM	79.12/94.20/86.00	91.27/93.50/92.37	93.47/75.69/83.65	86.12

Table 3. precision/recall/F1score and accuracy of both using the LiDAR points and both LiDAR points and DIM points based on the point-based classification.

2) Normalize the weight of each sample and each sample here is a point.

$$w_i^{t+1} = w_i^t / (\sum_{i=1}^{n+m} w_i^t) \quad (9)$$

where w^t is the weight of the i -th sample at t -th step
 n is the number of samples in the DIM points set
 m is the number of samples in the LiDAR points set

3) The features of all samples in the LiDAR points and DIM points sets are combined to form an input to a weighted random forest model. where h_t is the weighted random forest model at the t -th step.

4) Calculate the model error rate (ε_t). Because the TrAdaboost algorithm is used to classify the LiDAR points, DIM points do not participate in the calculation.

$$\varepsilon_t = \sum_{i=n+1}^{n+m} \frac{w_i^t |h_t^k(\mathbf{x}_i) - 1|}{\sum_{i=n+1}^{n+m} w_i^t} \quad (10)$$

In the formula, k represents the label of the sample \mathbf{x}_i . $h_t^k(\mathbf{x}_i)$ is the probability of \mathbf{x}_i to be the label k calculated by h_t .

5) Set β_t and β as the weights of h_t for samples in LiDAR points and DIM points, respectively. To avoid overfitting, in iterations, $\beta_t \leq 1/2$.

$$\beta_t = \varepsilon_t / (1 - \varepsilon_t) \quad (11)$$

$$\beta = 1 / (1 + \sqrt{2 \ln t / T}) \quad (12)$$

where T is the total number of iterations, and $T = 20$ in this paper.

6) Update w .

$$w_i^{t+1} = \begin{cases} w_i^t \beta_t^{|h_t^k(\mathbf{x}_i) - 1|}, & 1 \leq i \leq n \\ w_i^t \beta_t^{-|h_t^k(\mathbf{x}_i) - 1|}, & n+1 \leq i \leq n+m \end{cases} \quad (13)$$

7) Calculate the total probability by the second half of the random forest classifiers.

$$probability = \prod_{t=\lceil N/2 \rceil}^N \beta_t^{-h_t^k(x)} \quad (14)$$

The output is the final probability of each class, and a sample is labeled by the class with the highest probability.

From step 4 and step 7, it can be found that our method can classify the data into multiple categories. From step 6, it can be seen that if some samples of the DIM points are misclassified, the weights of the samples will be smaller in the next iteration. After multiple iterations, the weights of the samples that cannot help the point cloud to improve accuracy will be smaller and smaller, and the samples that can help the LiDAR points to improve accuracy will be retained. Therefore, the multi-class TrAdaboost algorithm can be used to fuse the two kinds of point cloud.

2.3 Super-voxel-based classification

Super-voxel-based approaches have been proposed in many present studies (Akwensi et al., 2020) and show better results than point-based classification. To test the generation of our framework, the super-voxel approach is also been considered here. The constructions of super-voxel are based on the study (Kang et al., 2018b).

First, the bounding box for the whole point cloud is obtained. Then, the bounding box is divided into small voxels whose sizes are $0.4\text{m} \times 0.4\text{m} \times 0.4\text{m}$. After the voxels have been generated, a super-voxel is generated by a three-dimensional simple linear iterative clustering algorithm. Thus first, all the voxels containing points are found and the points which are the closest to the centers of the voxels are used as seeds. The points whose distances to a seed are smaller than 0.8 m are seen as the neighboring points. If a point is the neighbor points of more than one seed, the point belongs to the closest seed. After all the points are assigned to the seeds, the seed position is updated. The point closest to the center of the neighbor points of a seed is the new seed. If the seeds are closer than 0.2m, the seeds are combined to one seed. After ten iterations, the point clouds are segmented to super-voxel.

After super-voxels have been done for the DIM points and the LiDAR points, the features of the super-voxel are extracted. The points in the super-voxel are used as the support region to calculate the F_v except for N_H . We use the average normalized height of the points in the super-voxel as the height feature N_H of the super-voxel. The corresponding F_v are the features of the super-voxel. Each feature of a super-voxel is a sample in the multi-class TrAdaboost algorithm. After the super-voxel of the LiDAR points have been classified, all the points in each super-voxel are given the same label with the super-voxel.

3. EXPERIMENTS AND RESULTS

3.1 Dataset Description

The LiDAR points and DIM points used in this paper were obtained by the WHU Kylin Cloud-I system which was developed by Professor Yang Bisheng's team at Wuhan University. The system was equipped with a Velodyne 16-line laser scanner and a consumer video camera. The scene is the playground of Wuhan University. The datasets obtained for this classification task are as shown in Figure 3. The three datasets

contained ground, buildings with different sizes, isolated trees, and clusters of vegetation. The detailed statistics of the two kinds of point clouds are as shown in Table 1.

In this paper, we divide the dataset into three groups for the experiment. Table 2 shows the train set and the test set we used for each experiment.

3.2 Results and Analysis

In order to evaluate the performance of the classification, four quantitative indexes are used, which are precision, recall, accuracy, and F1 score.

$$precision = \frac{tp}{tp + fn} \quad (15)$$

$$recall = \frac{tp}{tp + fp} \quad (16)$$

$$overall\ accuracy = \frac{tp + tn}{tp + fp + tn + tf} \quad (17)$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (18)$$

where tp, fp, tn, and fn represent true positives, false positives, true negatives, and false negatives in the confusion matrix, respectively

The experiments were performed by point-based, and super-voxel-based classification approaches, and the results are as shown in Figure 4, and 5, respectively. The points on trees, buildings, and ground are colored in red, blue, and green respectively. The quantitative results are listed in Table 3, and 4. It can be found in Table 3, and 4, the performance values obtained by using both the LiDAR points and the DIM points is good. And the accuracy attained by using our method is all higher than 80%. Among the three different categories, the accuracy of grounds and trees is high. This is mainly because the geometric characteristics of the two classes are more obvious. However, the accuracy of points on buildings is low. The main error areas in the three datasets are the top part of buildings, as well as the edges of the building. It can be seen from Figure 4, and 5, that many of the building points are misclassified to be vegetation. The main reason is that the distributions of points in the two places are between clutter and regular planes, which are easily confused.

In comparison to the results of only LiDAR points, most of the F1 scores and accuracies of each class using both the LiDAR points and DIM points listed in Table 3, and 4 achieve better results. The main misclassified areas are highlighted by yellow boxes. The results show our framework can effectively fuse the two kinds of point clouds, and the DIM points are helpful in the LiDAR points classification. It was observed that, after the fusion process, the overall precision of the vegetation class increased but that of the building class decreased. This is because the laser can penetrate the vegetation and there are points under the canopies of trees which are non-vegetation. However, the DIM points only located on the canopies. The two distributions are quite different. The DIM points provide more complete surface morphological information for the LiDAR points, so they are good supplements. For buildings,

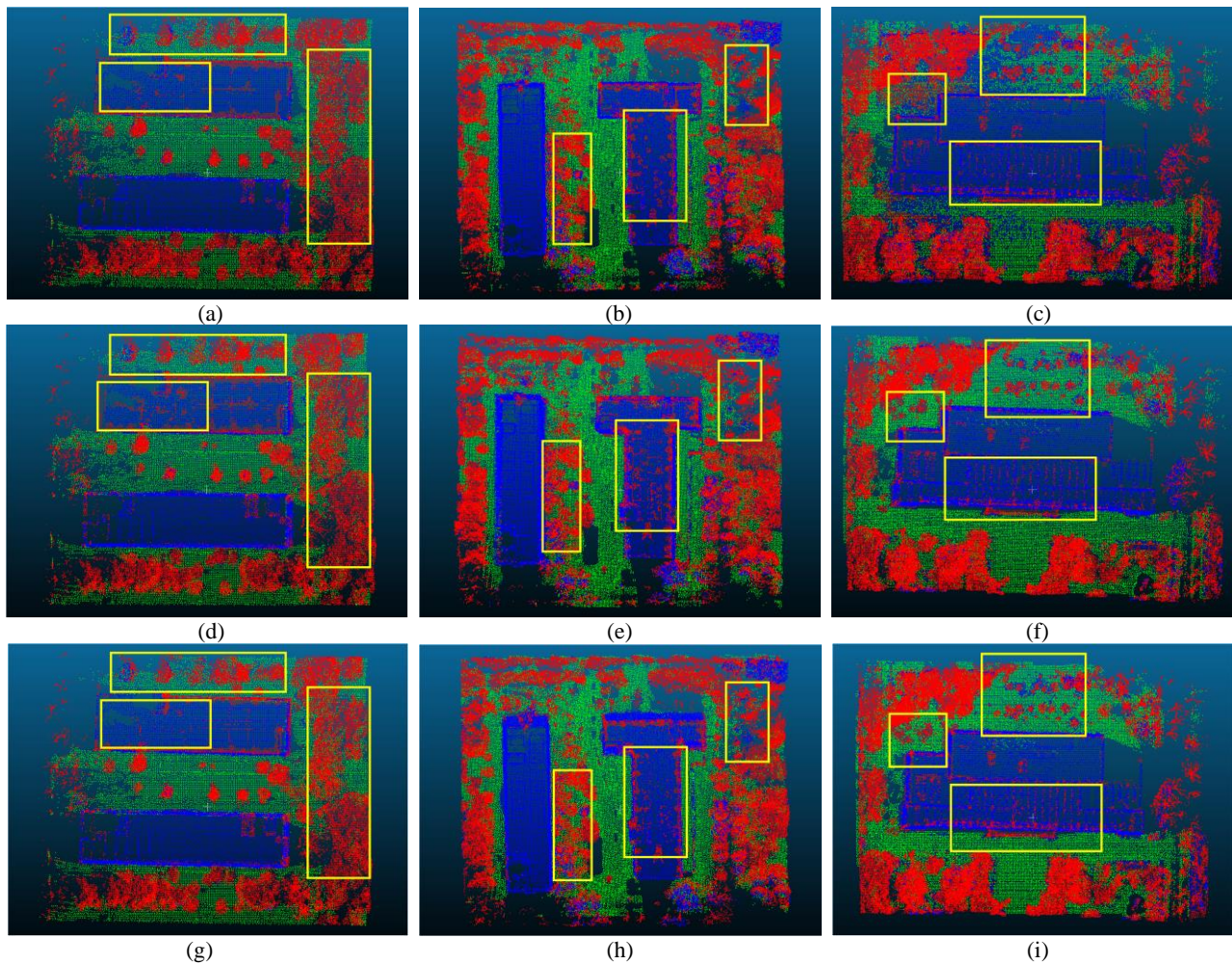


Figure 5. Super-voxel-based classification results. (a) - (c) are the results by using LiDAR points for scene II, III and I. (d) - (f) are the results by using LiDAR points and DIM points at the same regions for scene II, III and I. (g) - (i) are the results by using LiDAR points and DIM points in the different regions for scene II, III and I. Buildings, ground, and vegetation are colored in blue, green, and red.

category		Building(%)	ground(%)	vegetation(%)	Accuracy(%)
Test 1, Scene II	LiDAR	76.04/96.99/85.24	99.76/88.58/93.84	96.87/88.34/92.41	90.61
	LiDAR + DIM (the same regions)	74.03/98.38/84.48	99.42/92.00/95.56	98.42/86.60/92.14	90.63
	LiDAR + DIM (different regions)	75.46/96.96/84.87	99.31/92.05/95.54	97.68/87.24/92.16	90.72
Test 2, Scene III	LiDAR	76.79/81.40/79.03	97.81/92.89/95.29	89.50/88.44/88.97	87.43
	LiDAR + DIM (the same regions)	76.08/83.22/79.49	94.33/94.41/94.37	91.77/87.22/89.44	87.62
	LiDAR + DIM (different regions)	79.27/83.09/81.13	94.00/94.87/94.43	91.26/88.46/89.84	88.25
Test 3, Scene I	LiDAR	82.29/69.77/75.56	58.02/99.45/73.28	95.75/85.61/90.40	82.95
	LiDAR + DIM (the same regions)	81.41/85.89/83.59	87.74/98.06/92.61	95.36/88.37/91.73	89.96
	LiDAR + DIM (different regions)	78.94/83.93/81.36	82.73/99.10/90.18	96.49/86.71/91.34	88.65

Table 4. Precision/recall/F1 score and accuracy of both using the LiDAR points and both LiDAR points and DIM points (at the same regions and different regions) based on the super-voxel-based classification approach

the places where most of the points are misclassified are place with the lack of texture in the image. There is data missing at these places, so the DIM points are not useful in complementing the buildings. However, comprehensive

consideration of the precision and recall, and the F1 score of the two classes are still shown improvements. For the ground in scene II and III, most points on the ground are classified right, so the effect of fusion is not obvious. In the scene I, the

category		Building(%)	ground(%)	vegetation(%)	Accuracy(%)
Test 1, Scene II	The proposed method	74.03/98.38/84.48	99.42/92.00/95.56	98.42/86.60/92.14	90.63
	SVM	72.87/95.79/82.77	99.57/87.69/93.25	96.25/86.81/91.29	89.24
	Adaboost	54.89/96.81/70.06	100/87.34/93.24	96.73/78.19/86.48	83.67
Test 2, Scene III	The proposed method	76.08/83.22/79.49	94.33/94.41/94.37	91.77/87.22/89.44	87.62
	SVM	71.53/76.26/73.53	96.05/91.78/93.87	86.73/85.21/85.96	84.13
	Adaboost	57.29/90.50/70.16	99.17/90.92/94.87	95.91/81.03/87.84	85.05
Test 3, Scene I	The proposed method	81.41/85.89/83.59	87.74/98.06/92.61	95.36/88.37/91.73	89.96
	SVM	68.40/84.57/75.63	89.15/98.80/93.86	95.43/83.44/89.04	87.15
	Adaboost	55.31/46.30/50.40	38.09/99.33/55.07	96.58/79.69/87.33	71.59

Table 5. Precision/recall/F1 score and accuracy between different classifiers based on the supervoxel-based classification

performance of the ground is significantly improved. It is because in the ground of scene III, the LiDAR points have a missing part due to building occlusion, and the DIM points make up for this part. Fusion with the kind of DIM points will improve the classification performance.

To further evaluate the performance of our proposed model, frequently-used classifiers (Adaboost (Wang et al., 2015), SVM (Ramiya et al., 2016)) were compared to the proposed model using the three datasets. Table 5 lists the comparison results of the three datasets and Table 6 indicates the processing time. Although the proposed method required more processing time than other classifiers, it achieved the highest overall accuracies in terms. Furthermore, compared with other classifiers, the proposed model has better precision in the classification of buildings.

For the fusion of optical images and point cloud, the two data sources need to be registered. However, in our frameworks, only samples are used, which means no registration is needed. A big advantage of no registration is that the DIM points in other regions can be used to improve the LiDAR points in the region. This decreases the requirement of images in the same area during the fusion process. To show the performance, DIM points in another region is used as auxiliary data. Figure 6 shows the DIM points of a village in Zhengding City, Hebei Province obtained by a UAV with the Nikon D300 camera. Buildings, vegetation, and grounds are also contained in the DIM points. As the super-voxel classification approach obtain the best results in the fore experiments, here only the approach is used. Table 4 shows the results. It can be seen that after fusion with the DIM points in different regions, the classification performances are still better than those using the LiDAR points alone. But lower than fusion with the DIM points in the same regions. This is because the environment is different, the DIM points in Hebei cannot be well assisted.

However, in scene I, the precision of vegetation is higher than using DIM points in the same regions. Because in scene III, the DIM points of vegetation have missing, and the vegetation points in Hebei are more complete, which can assist the classification of LiDAR points better. It can be found that, the better the auxiliary point clouds are, the better the classification performance achieved.

4. CONCLUSION

In this paper, we show that DIM points are helpful for the LiDAR points classification, which is another way of fusing point clouds and images. A sample selection framework is

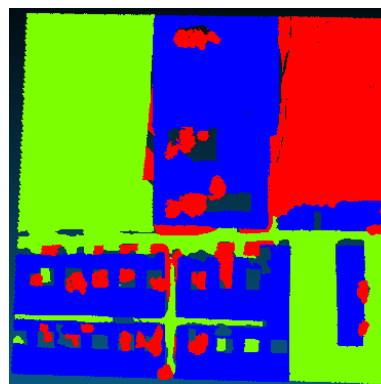


Figure 6 DIM points of a village in Zhengding City, Hebei Province

Method		The proposed method	SVM	Adaboost
Time (second)	Test 1, Scene II	162.22	104.53	16.16
	Test 2, Scene III	106.38	72.96	13.34
	Test 3, Scene I	183.96	180.89	16.74

Table 6. The processing time of different classifiers

proposed to fuse the DIM points. In the framework, the features of points, and super-voxels are extracted. Then the multi-class TraAdaboost algorithm was proposed to automatically select samples from the DIM points for improving the classification performance of the LiDAR points. The results show that after using the DIM points, the classification performance improved and the improvement is based on the quality of the DIM points. At the same time, the framework does not need to register the LiDAR points and DIM points. The DIM points of other regions can also help in the LiDAR points classification. This advantage can make full use of existing aerial images.

In the future, we will use our framework for classifying more objects, and study which categories can be improved by the DIM points. Also we will use more DIM points for classification to further study which DIM points can improve the LiDAR points.

REFERENCES

Ahmad K. A., Paul C., 2014: Laurent Trassoudaine. Super-Voxel Based Segmentation and Classification of 3D Urban

- Landscapes with Evaluation and Comparison. *Springer Tracts in Advanced Robotics*, 92, 511-526.
- Akwensi P. H., Kang Z., Yang J., 2020: Fisher Vector Encoding of Supervoxel-Based Features for Airborne LiDAR Data Classification. *IEEE Geoscience and Remote Sensing Letters* 17(3): 504-508 (2020)
- Blei D. M., Ng A. Y., Jordan M. I., 2003: Latent Dirichlet allocation. *J. Mach. Learn. Res.*, 993–1022.
- Cao Y., Wei H., Zhao H., 2012: An effective approach for land-cover classification from airborne lidar fused with co-registered data. *International Journal of Remote Sensing*, 33(18), 5927-5953.
- Chehata N., Guo, Mallet C., 2009: Airborne lidar feature selection for urban classification using random forests. *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, 38, 207 – 212.
- Chen R., Chen S., Yang L., 2019: Fault diagnosis of rolling bearings based on improved TrAdaBoost multi-classification algorithm. *Journal of Vibration and Shock*, 38 (15), 36-41,48.
- Dai W., Yang Q., Xue G. R., 2007: Boosting for transfer learning. *International Conference on Machine Learning. New York:ACM*, 193-200
- Fan S., Zhang A., Hu S., 2013: Classification method of airborne laser full waveform point cloud data based on random forest. *China laser*, 2013 (09): 221-227.
- Feng S., 2014: Comparison of image matching point cloud and airborne laser point cloud. *Geospatial Information*, (6), 82-83.
- Guo L., Chehata N., Clément M., 2011: Relevance of airborne lidar and multispectral image data for urban scene classification using Random Forests. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(1), 56-66.
- Haala N., Brenner C., 1999: Extraction of buildings and trees in urban environments. *Isprs Journal of Photogrammetry & Remote Sensing*, 54(2–3), 130-137.
- He X., Zou Z., Zhang Y., 2018: Object-oriented classification method for oblique photogrammetric point clouds. *Remote Sensing for Land and Resources*, 117 (02), 90-95.
- Huang J., You S., 2016: Point Cloud Labeling using 3D Convolutional Neural Network International Conference on Pattern Recognition.
- Kang Z., Yang J., Zhong R., Wu Y., Shi Z, Lindenbergh R., 2018a: Voxel-Based Extraction and Classification of 3-D Pole-Like Objects From Mobile LiDAR Point Cloud Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11. 4287-4298. 10.1109/JSTARS.2018.2869801.
- Kang, Z., Yang, J., 2018b: A probabilistic graphical model for the classification of mobile LiDAR point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*. 143. 16. 10.1016/j.isprsjprs.2018.04.018.
- Niemeyer J., Rottensteiner F., Soergel U., 2012: Conditional Random Fields for Lidar Point Cloud Classification in Complex Urban Areas. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences I-3*, 263-268.
- Niemeyer J., Rottensteiner F., Soergel U., 2013: Classification of urban LiDAR data using conditional random field and random forests. *Urban Remote Sensing Event. IEEE* 139-142.
- Ramiya A. M., Nidamanuri R. R., Ramakrishnan K., 2016: A supervoxel-based spectro-spatial approach for 3D urban point cloud labelling. *International Journal of Remote Sensing*, 37(17), 4172-4200.
- Rosnell T., Honkavaara E., 2012: Point Cloud Generation from Aerial Image Data Acquired by a Quadcopter Type Micro Unmanned Aerial Vehicle and a Digital Still Camera. *Sensors*, 12(12), 453-480.
- Sampath A., Shan J., 2010: Segmentation and Reconstruction of Polyhedral Building Roofs From Aerial Lidar Point Clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 48(3), 1554-1567.
- Sohn G., Huang X., Tao V., 2008: Using a Binary Space Partitioning Tree for Reconstructing Polyhedral Building Models from Airborne Lidar Data. *Photogrammetric Engineering & Remote Sensing*, 74(11), 1425-1438.
- Sun J., Lai Z., 2014: Airborne LiDAR feature selection for urban classification using random forests. *Wuhan Daxue Xuebao (Xinxi Kexue Ban)/Geomatics and Information Science of Wuhan University*, 39(11), 1310-1313.
- Thiel C., Schmullius C., 2017: Comparison of UAV photograph-based and airborne lidar-based point clouds over forest from a forestry application perspective. *International Journal of Remote Sensing*, 38(8-10), 16.
- Wang, Z., Zhang, L., Fang, T., Mathiopoulos, P., Tong, X., Qu, H., Xiao, Z., Li, F., Chen, D., 2015: A multi-scale and hierarchical feature extraction method for terrestrial laser scanning point cloud classification. *IEEE T. Geosci. Remote. Sens.* 53 (5), 2409-2425.
- Weinmann M , Jutzi B , Mallet C, 2014: Semantic 3D scene interpretation: A framework combining optimal neighborhood size selection with relevant features.
- Yang J., Yu K., Gong Y., Huang T., 2009: Linear spatial pyramid matching using sparse coding for image classification, *Proc. IEEE CVPR*, 1794-1801.
- Zhang Z., Zhang L., Tong X., 2016: A Multilevel Point-Cluster-Based Discriminative Feature for ALS Point Cloud Classification. *IEEE Transactions on Geoscience & Remote Sensing*, 54(6), 1-13.
- Zhao L., Wang H., Guo Z., 2018: A study of classification of point clouds generated by oblique imagery based on random forest. *Surveying and Mapping Engineering*, 27 (12), 16-21.