The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-2/W13, 2019
ISPRS Geospatial Week 2019, 10–14 June 2019, Enschede, The Netherlands

A Modified Three-Dimensional Gray-Level Co-occurrence Matrix For Image Classification With Digital Surface Model

L. Yan, W. Xia*

School of Geodesy and Geomatics, Wuhan University, 129 Luoyu Road, Wuhan 430079, China;
lyan@sgg.whu.edu.cn (L.Y.); xiawang@whu.edu.cn (W.X.)

KEY WORDS: GLCM, Feature Extraction, Random Forest, Image Classification, Digital Surface Model, Image Texture analysis

ABSTRACT:

2D texture cannot reflect the 3D object's texture because it only considers the intensity distribution in the 2D image region but int real world the intensities of objects are distributed in 3D surface. This paper proposes a modified three-dimensional gray-level co-occurrence matrix (3D-GLCM) which is first introduced to process volumetric data but cannot be used directly to spectral images with digital surface model because of the data sparsity of the direction perpendicular to the image plane. Spectral and geometric features combined with no texture, 2D-GLCM and 3D-GLCM were put into random forest for comparing using ISPRS 2D semantic labelling challenge dataset, and the overall accuracy of the combination containing 3D GLCM improved by 2.4% and 1.3% compared to the combinations without textures or with 2D-GLCM correspondingly.

1. INTRODUCTION

Semantic classification of remote sensing images plays an important role for a wide range of applications and is still a challenging problem (Cheng et al. 2016). With the advance of feature representations and classifiers, machine learning-based methods have achieved significant improvement. Feature extraction and classifier training are major steps in machine learning, and sometimes feature fusion (Wang et al. 2013) and dimension reduction (Hariharan et al. 2012) are still needed.

Features can be categorized as handcraft features defined by experts or features learnt from samples by deep learning which are popular in recent years (Paisitkriangkrai et al. 2015). However, the features learnt by deep learning are not intuitional, which makes it difficult for experts to understand the scene.

Handcraft features can be categorized as spectral features, geometric features or textures. Spectral features only consider spectral information, geometric features only consider geometric information while textures consider both.

The development of acquisition and processing technology makes it easy to get 3D information of earth surface, like point cloud or DSM (Qin et al. 2016; Gerke, 2014). Many researchers created 3D geometric features especially in point cloud classification. Geometric features reflect objects' local shape, height and more complex structure. Local shape features (Pauly et al. 2003) are presented by normals, linearity, planarity, sphericity, surface variation and so on. Complex structure features (Weinmann et al. 2018) are combination of complementary features.

There is no exact definition of the texture, but it can be seen as the variation or repetition of spectral values in space. One type of textures are pattern features which are computed by placing primitives in local image regions and analysing the relative differences (Li et al. 2015), like Gabor features (Jain et al. 1997) and Local binary patterns (LBP) (Ojala et al. 2002). Another type of textures is computed from probability distribution of local image attributes (Li et al. 2015) like entropy-based saliency (Kadir et al. 2001) and co-occurrence features (Conners et al. 1984, Muller et al. 2001, Palm 2004).

Most textures were created for 2D image regions, while there were rare studies for 3D textures. 2D textures cannot represent the real textures of 3D objects in 3D space. Textures reflect variation or repetition of spectral values in space, and relative positions of primitives in 2D image region cannot reflect their real relation in 3D space. Hence, 3D textures will be more robust to variation of view angles.

In medical image processing, 3D gray-level co-occurrence matrix (3D-GLCM) were used of retrieval (Qian et al. 2011) or recognition (Kurani et al. 2004, Chen et al. 2009). However, it cannot be used directly to land-cover classification. Spatial relation is simple in 3D medical images whose 3D data is represented by the image stack. It means that every pixel of an image can find an upper or lower pixel unless the pixel is in the upmost or lowermost image. In remote sensing area, 3D data is sparse in the third dimension and is normally represented as point cloud or spectral image with the corresponding depth image like RGB-D image or DOM with DSM. The relation of pixel/point positions is irregular.

This paper solved the problem of the computation of 3D-GLCM for DOM with DSM considering the data sparsity of the direction perpendicular to the image plane. To evaluate the effectiveness of 3D-GLCM, different combinations of different features were put into random forest for semantic classification and comparison.

2. METHOD

To validate the effectiveness of the proposed features, different feature combinations are used for comparing and random forest is used as classifier.

2.1 3D Gray-Level Co-occurrence Matrix

After a brief introduction of the original 2D-GLCM, the GLCM for the image stack in the medical area was introduced. Then, a modified version of 3D-GLCM was proposed considering the data sparsity of the direction perpendicular to the image plane.

To avoid confusion, 3D-GLCM only means the method proposed in this paper, and corresponding method in medical areas will be called volumetric data GLCM (**VD-GLCM**) since it can only process volumetric data.

* Corresponding author

2.1.1 2D GLCM

2D-GLCM (Haralick et al. 1973) consider not only the distribution of spectral values but also the relative positions of pixels in a 2D image region. Through computing the correlation between intensities of two pixels with a certain distance and direction, GLCM can reflect comprehensive information of the image region.

GLCM is a $L \times L$ matrix, and L is intensity levels. The element of matrix is the count of intensities of pixel pairs with a certain distance and direction. Let $G_p(i,j,d,\theta)$ represent 2D-GLCM of the pixel p with distance d and direction θ, (i,j) is the row and column number of the matrix. Let P^2 represent all the pixel pairs in the window, the 2D-GLCM can be described in the following equation:

$$G_p(i,j,d,\theta) = \#\left\{(p_1,p_2) \in P^2 \middle| \begin{array}{c} I(p_1)=i, I(p_2)=j \\ Che\_dis(p_1,p_2)=d \\ \Theta(p_1,p_2)=\theta \end{array}\right\} \quad (1)$$

where $I(p)$ is the intensity of pixel p, $Che\_dis(p_1,p_2)$ and $\Theta(p_1,p_2)$ are Chebyshev distance (Klove et al. 2010) and planer direction between p_1,p_2, and $\#$ is the operator that counts the number in the set. Chebyshev distance (Baeza-Yates et al. 2002) is a metric defined on a vector space where the distance between two vectors is the greatest of their differences along any coordinate dimension. Pixel pairs with a certain distance and direction can be represented by a displacement vector. Table 1 lists the displacement vectors for 2D-GLCM.

Direction ($\boldsymbol{\theta}$)	**Displacement Vector**
0°	$(d,0)$
45°	(d,d)
90°	$(0,d)$
135°	$(-d,d)$

Table 1. Displacement Vectors for GLCM for 2D Data

Direction $(\theta,\emptyset)$	**Displacement Vector**
$(\mathbf{0^\circ},\mathbf{90^\circ})$	$(d,0,0)$
$(\mathbf{45^\circ},\mathbf{90^\circ})$	$(d,d,0)$
$(\mathbf{90^\circ},\mathbf{90^\circ})$	$(0,d,0)$
$(\mathbf{135^\circ},\mathbf{90^\circ})$	$(-d,d,0)$
$(\sim,\mathbf{0^\circ})$	$(0,0,d)$
$(\mathbf{0^\circ},\mathbf{45^\circ})$	$(d,0,d)$
$(\mathbf{45^\circ},\mathbf{90^\circ}-\mathbf{arcsin}\sqrt{\mathbf{1/3}})$	(d,d,d)
$(\mathbf{90^\circ},\mathbf{45^\circ})$	$(0,d,d)$
$(\mathbf{135^\circ},\mathbf{90^\circ}-\mathbf{arcsin}\sqrt{\mathbf{1/3}})$	$(-d,d,d)$
$(\mathbf{0^\circ},\mathbf{135^\circ})$	$(d,0,-d)$
$(\mathbf{45^\circ},\mathbf{90^\circ}+\mathbf{arcsin}\sqrt{\mathbf{1/3}})$	$(d,d,-d)$
$(\mathbf{90^\circ},\mathbf{135^\circ})$	$(0,d,-d)$
$(\mathbf{135^\circ},\mathbf{90^\circ}+\mathbf{arcsin}\sqrt{\mathbf{1/3}})$	$(-d,d,-d)$

Table 2. Displacement Vectors for GLCM for Volumetric Data

2.1.2 VD-GLCM for the image stack

The image stack is a 3D volume data which consists of a series of images with the same size. The key of VD-GLCM is to compute the relative position in the 3D region. The main difference between 2D and VD GLCM is the way to determine the direction. For 2D GLCM, the direction is planar directions. Normally, the direction angles are 0°, 45°, 90°, 135°. There are more directions in VD-GLCM, considering pixels of upper and lower images. Let P^2 represent all the pixel pairs in the volume data, the VD-GLCM can be described in the following equation:

$$G_p(i,j,d,\theta,\emptyset) = \#\left\{(p_1,p_2) \in P^2 \middle| \begin{array}{c} I(p_1)=i, I(p_2)=j \\ Che\_dis(p_1,p_2)=d \\ \Theta(p_1,p_2)=\theta \\ \Phi(p_1,p_2)=\emptyset \end{array}\right\} \quad (2)$$

where $\Phi(p_1,p_2)$ is vertical direction between p_1,p_2, and other notations are the same with notations in equation (1). Table 2 lists the displacement vectors for VD-GLCM. It should be noted that vertical direction is described as zenith angle which is the included angel between the vector and the direction perpendicular to the image plane.

2.1.3 3D-GLCM for Grayscale image with Depth Image

VD-GLCM cannot process the grayscale image with the depth image because of its sparsity and irregularity in the vertical direction. In volumetric data, distance and vertical directions of pixel pairs are fixed as showed in Table 2. However, for depth image, vertical directions are unfixed and can be any angle in the range $[0^\circ,180^\circ)$ and the distance is unfixed too.

To count pixel pairs for 3D-GLCM, the vertical direction range are divided into N sections, and the range of i-th section Sec_i is:

$$Sec_i = \left[\frac{180^\circ}{N}\cdot(i-1), \frac{180^\circ}{N}\cdot i\right) \quad (3)$$

In this paper, N is set to 4, so there will be 4 sections which are $[0^\circ,45^\circ)$, $[45^\circ,90^\circ)$, $[90^\circ,135^\circ)$, $[135^\circ,180^\circ)$. Every section is seen as a vertical direction and hence the directions of pixel pairs in 3D data are fixed. Besides, only horizonal distance on image plane is considered. Let P^2 represent all the pixel pairs in the 3D data, the 3D-GLCM can be described in the following equation:

$$G_p(i,j,hd,\theta,Sec_i) = \#\left\{(p_1,p_2) \in P^2 \middle| \begin{array}{c} I(p_1)=i, I(p_2)=j \\ HChe\_dis(p_1,p_2)=hd \\ \Theta(p_1,p_2)=\theta \\ \Phi(p_1,p_2)=Sec_i \end{array}\right\} \quad (4)$$

where $HChe\_dis(p_1,p_2)$ is horizonal Chebyshev distance between p_1,p_2, which means that it computes greatest of differences between p_1,p_2 along planer dimensions without vertical dimension.

The generation process of 3D-GLCM is different with 2D-GLCM and VD-GLCM, which should determine the distance and direction before counting pixel pairs. For 2D-GLCM, only horizonal distance and direction are determined before counting, and vertical direction will be computed in the counting process. The specific generation process of 3D-GLCM is in the following:

1. Determine the image window size of pixel P and the intensity levels L.

2. Determine the horizonal distance hd and horizonal direction θ.
3. Divide the vertical direction into N sections according to equation (3), and prepare N matrix with size of $L \times L$.
4. In the window of P, find all the pixel pairs satisfied the condition in process 2.
5. Compute the vertical direction of every pixel pair in process 3. The pixel pair will be counted in the corresponding matrix according to its vertical direction.
6. N GLCM will be generated in the meantime, which count the same planar direction but different vertical directions.

Every pixel will produce $4 \times$ N GLCM matrix. Then, Haralick features (Kurani et al. 2004) will be computed for quantitatively describing GLCM.

2.2 Semantic Classification

To evaluate the effectiveness of 3D-GLCM in semantic classification, different features are used and combined. Random forest is used as classifier. In our work, we assume that the spectral bands used for the orthophoto comprise the near-infrared (NIR), red (R) and green (G) bands (Cramer, 2010; Rottensteineret al., 2012; Gerke, 2014).

2.2.1 Feature Combination

Textures combined with spectral and geometric features are used for comparison. Spectral features are normalized color and NDVI. Normalized color (Gevers et al. 1999) is used to improve robustness with respect to changes in illumination. NDVI (Rouse et al. 1973) is a strong indicator for vegetation.
Geometric features are normalized DSM (nDSM) (Gerke, 2014) and 3D structure tensor (Weinmann et al. 2018) which can be used to calculate the features of linearity, planarity, sphericity, omnivariance, anisotropy, eigenentropy and change of curvature (Pauly et al. 2003, West et al. 2004).
Textures are 2D and 3D GLCM with energy as the GLCM feature (Conners et al. 1984, Muller et al. 2001). Since GLCM only processes grayscale images, the spectral bands of images were merged into one band by calculating their means.
Three different combination of features were put in random forest for comparing. First combination only used spectral and geometric features, second combination added 2D GLCM into the first combination and third combination add 3D GLCM in to the first combination.

2.2.2 Random Forest

Random forests (Genuer et al. 2010) are an ensemble method (Maclin et al. 1999, Rokach 2010) and combines a multitude of decision trees (Holzinger 2015), using randomly subsets of training samples. Random forest is well-used in remote sensing areas (Belgiu et al. 2016) because of its improvement of generalizability and robustness over a single decision tree.
Another advantage of random forests is that they can generate features importance of input features, which can be used to analyze features' effectiveness (Genuer et al. 2010, Louppe 2014).

2.2.3 Accuracy Assessment

To analyse the effectiveness of different features, some accuracy features are used to quantitatively evaluate the classification result for each category as well as the overall categories.
The evaluation of the classification ability for each class is like the evaluation for a binary classification task. The terms 'positive' and 'negative' refer to the classifier's prediction, and the terms 'true' and 'false' refer to whether that prediction corresponds to the observation. Let TP_i (true positive) represent the number of pixels which are correctly classified of i-th category, FP_i represent the number of pixels which are wrongly classified of i-th category and TN_i represent the number of pixels which are missed of i-th category, then we can have the notions in the following:

$$prediction_i = \frac{TP_i}{TP_i + FP_i} \tag{5}$$

$$recall_i = \frac{TP_i}{TP_i + FN_i} \tag{6}$$

$$F1\_score_i = 2 \times \frac{prediction_i \times recall_i}{prediction_i + recall_i} \tag{7}$$

To evaluate the classification of overall categories, the overall accuracy is calculated. Let of K be the number of categories, N be the number of all pixels and i be the i-th category, then we can have:

$$overall\_accuracy = \frac{\sum_i^K TP_i}{N} \tag{8}$$

Features		Spec+Geo	Spec+Geo+ 2D-GLCM	Spec+Geo+ 3D-GLCM
Impervious Surfaces	pre	0.885	0.892	0.892
	rec	0.823	0.843	0.875
	F1	0.853	0.867	0.883
Building	pre	0.945	0.949	0.95
	rec	0.907	0.915	0.92
	F1	0.926	0.932	0.935
Low Vegetation	pre	0.774	0.784	0.799
	rec	0.755	0.762	0.768
	F1	0.764	0.773	0.783
Tree	pre	0.847	0.847	0.853
	rec	0.866	0.875	0.886
	F1	0.856	0.861	0.869
Car	pre	0.196	0.24	0.32
	rec	0.788	0.802	0.77
	F1	0.314	0.369	0.452
Overall Accuracy		0.841	0.852	0.865

Table 3. The F1-scores for five classes and the overall accuracy. 'Spec' means spectral features, 'Geo' means geometric features, 'pre' means prediction, 'rec' means recall and 'F1' means F1 score.

3. EXPERIMENTATION

3.1 Material

The proposed method was applied to the ISPRS Vaihingen dataset (Cramer 2010; Rottensteiner et al., 2012). The dataset contains 33 patches (of different sizes), and each patch consists of a true orthophoto (TOP) with near infrared, red and green bands and a DSM with a ground sampling distance of 9 cm. Labelled ground truth was also provided for 16 of the areas before summer 2018, and were made up of 6 categories which were showed in different colors in ground truth as follows:

1. Impervious surfaces (RGB: 255, 255, 255)
2. Building (RGB: 0, 0, 255)

https://doi.org/10.5194/isprs-archives-XLII-2-W13-133-2019 | © Authors 2019. CC BY 4.0 License.

3. Low vegetation (RGB: 0, 255, 255)
4. Tree (RGB: 0, 255, 0)
5. Car (RGB: 255, 255, 0)
6. Clutter/background (RGB: 255, 0, 0)

Although ground truth of all 33 patches were provided now, we still used 16 patches for training and validation and the left 17 patches for final test. 300000 training pixels for each class are chosen at random and used along with the corresponding ground truth pixel labels to train a random forest classifier with 75 trees. The Clutter/background class is excluded from this classifier.

Three kinds of features of each pixel were extracted. Spectral features were normalized color and NDVI, geometric features were nDSM and 3D structure tensor, and textures were 2D-GLCM and 3D-GLCM. Three combination of these features were used and combined as follows:

7. Spectral features + Geometric features
8. Spectral features + Geometric features + 2DGLCM
9. Spectral features + Geometric features + 3DGLCM

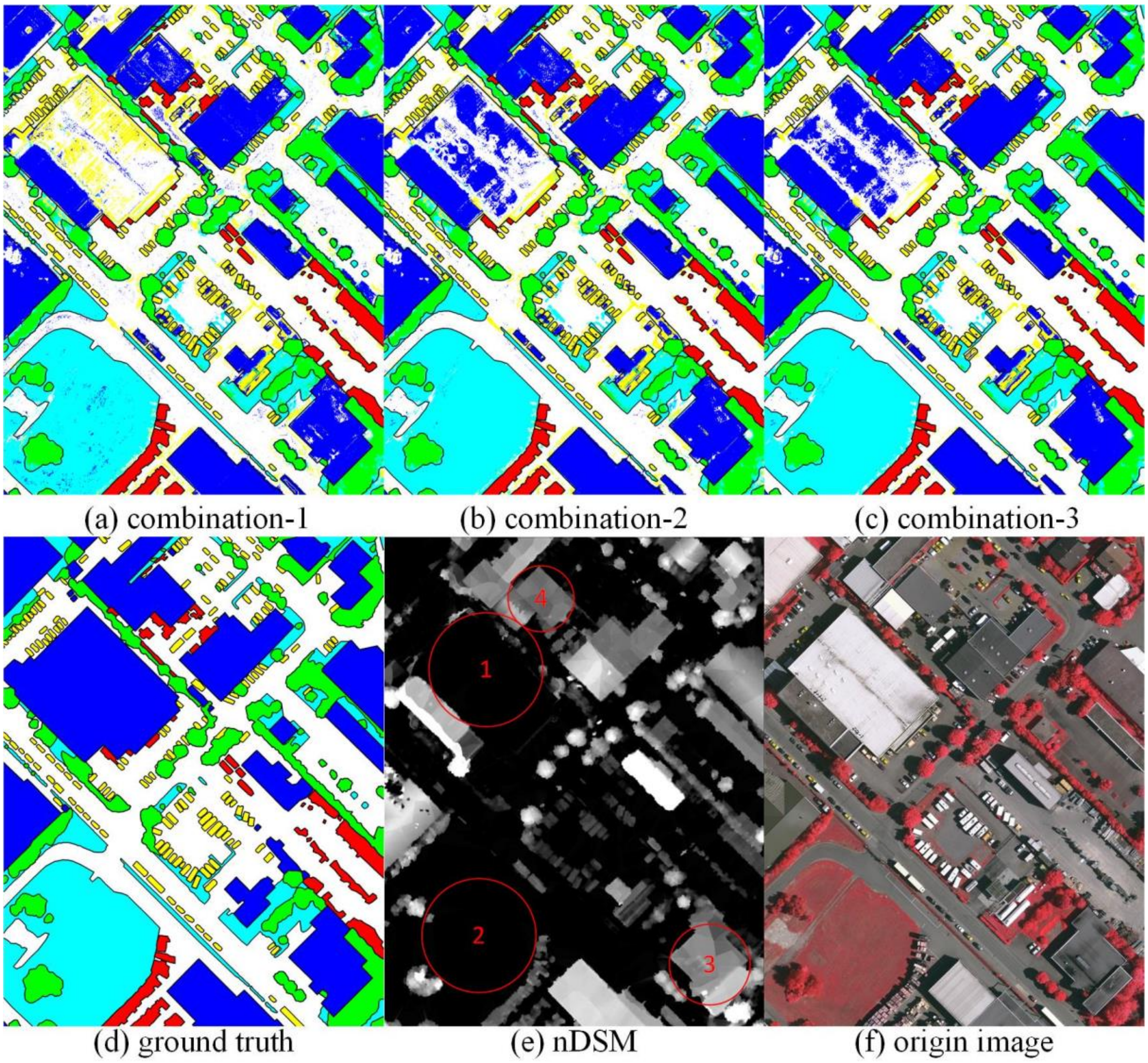

Figure 1. Classification results of 'area-31'. (a)(b)(c) are classification result of three combination of features; (d) is ground truth; (e) is nDSM; (f) is origin image with three RGB bands corresponding to the near infrared, red and green bands

3.2 Results

Table 3 shows the evaluation result for three combinations of features. Combination-2 gets higher prediction, recall and F1-score for every category than combination-1 except the prediction of tree, and the overall accuracy is higher too. It means that textures could effectively improve the identification of every kind of object.

Combination-3 get the highest overall accuracy and the highest F1-score for every category. Except the recall of car and the prediction of impervious surface, prediction and recall of every category of combination-3 is higher than combination-2. It means that 3D-GLCM had a better improvement of identification than 2D-GLCM.

Figure-1 shows the classification result of 'area-31' in Vaihingen dataset. Figure-1e is the nDSM of 'area-31' with four areas marked by red circles. In red circle-1, buildings had low pixel values in nDSM, which made it easy to be wrongly identified. Combination-1 missed the most of the building while combination-2 identified half of it. Combination-3 identified more pixels than combination-2 but the advantage is not very obvious.

In red circle-2, some low vegetation pixels were identified as building by combination-1 while the other two combination identified them correctly. In circle-3 and circle-4, some areas of buildings were identified as impervious surface by combination-1, while the other two combinations had better identification results and combination-3 was the best.

Besides, there were many small blued speckles on the roads in Figure-1a, and these blue speckles were less in figure-1b and almost reduced in figure-1c. It means that some impervious surface areas were identified as buildings by combination-1, while the other two combinations improved the results and combination-3 identified most of them correctly.

3.3 Discussion

Height is no doubt an important feature for 3D-classification, and nDSM (Gerke, 2014) is generated from DSM to represent the height of objects. However, there were some problems about nDSM. First, nDSM classified DSM into ground and off-ground pixels using lastools-toolbox(1), which might identify the building with big flat roof as ground as showed in circle-1 in Figure-1e. This would made it difficult to identify this building. What's more, the samples of the classifier were generated at random. If the pixels of this building were chosen as samples, the effectiveness of nDSM feature would be reduced and the low and high objects might be confused. Second, the nDSM values of the same building varied as showed in circle-3 and 4 in figure-1e. Hence, the generation of shape features and textures should use DSM which reflected the real relative height. Still, the detection of the buildings was influenced and many small speckles of buildings were identified as ground as showed in figure-1a.
The use of textures alleviated the problems introduced by nDSM. More than half of the building in circle-1 in figure-1 is identified by combination-2 and 3, and less speckles showed in circle-3 and 4. Combination-3 reduced more speckles and identified more building pixels than combination-2. Besides, blue speckles in circle-2 and on the roads in figure-1a were reduced in figure-1b and c, and combination-3 had a better performance on reduction. The introduction of textures could improve the identification of all categories as showed in table-3, and 3D-GLCM did an obviously better job than 2D-GLCM. The real world is 3D space, and intensities are distributed in 3D surface. 2D GLCM cannot reflect the real object texture because it only considered the intensity distribution in the 2D image region without height information. 3D GLCM counted the intensity pairs in 3D region, reflecting the real texture. Therefore, 3D GLCM had better performance and improve results for all the classes.

4. CONCLUSION

2D texture cannot reflect 3D objects' texture in real 3D world. There were rare researches about 3D-textures of classification or object extraction in remote sensing or photogrammetry area. 3D-GLCM was first proposed in the metical area but it cannot be used directly considering the data sparsity of the direction perpendicular to the image plane. To address this problem, a modified 3D-GLCM was proposed, dividing the vertical direction into a number of sections to avoid the data sparsity. The experiment on the ISPRS Vaihingen dataset showed that textures could effectively improve the classification result, and 3D-GLCM had a better improvement than 2D-GLCM.
There were some problems about the work in this paper. First, pixel-based classification using only random forest without constraint of context made many small speckles even with the help of 3D-GLCM. The main goal of this paper was to evaluate the effectiveness of 3D-GLCM and compare it with the traditional features. So, we did not want to introduce other factors on comparison, such as the segmentation problem of object-based methods (Ma et al. 2016), or many parameters setting like MRF (Kumar et al. 2003) or CRF (Chen et al. 2018). Second, we only used energy as the GLCM feature. The vertical direction was only divided into 4 sections in the experiment, and the impact of more sections was not discussed. These problems were worthy of being studied in the future.

5. ACKNOWLEDGEMENTS

This research was supported by The National Key Research and Development Program of China under grant no. 2016YFB0501403.
The Vaihingen data set was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer 2010): http://www.ifp.uni-stuttgart.de/dgpf/DKEP-Allg.html

6. REFERENCES

Baeza-Yates, R., A. Moffat and G. Navarro, 2002. Handbook of Massive Data Sets.

Belgiu, M. and L. Dragut, 2016. Random forest in remote sensing: A review of applications and future directions. *Isprs Journal of Photogrammetry and Remote Sensing*, **114**: 24-31. DOI: 10.1016/j.isprsjprs.2016.01.011.

Chen, L. C., G. Papandreou, I. Kokkinos, K. Murphy and A. L. Yuille, 2018. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *Ieee Transactions on Pattern Analysis and Machine Intelligence*, **40**(4): 834-848. DOI: 10.1109/Tpami.2017.2699184.

Chen, W. S., R. H. Huang and L. Hsieh, 2009. Iris Recognition Using 3D Co-occurrence Matrix. *Advances in Biometrics*, **5558**: 1122-+.

Cheng, G. and J. Han, 2016. A Survey on Object Detection in Optical Remote Sensing Images. *Isprs Journal of Photogrammetry & Remote Sensing*, **117**: 11-28.

Conners, R. W., M. M. Trivedi and C. A. Harlow, 1984. Segmentation of a High-Resolution Urban Scene Using Texture Operators. *Computer Vision Graphics and Image Processing*, **25**(3): 273-310. DOI: Doi 10.1016/0734-189x(84)90197-X.

Cramer, M., 2010. The DGPF-Test on Digital Airborne Camera Evaluation - Overview and Test Design. *Photogrammetrie Fernerkundung Geoinformation*,(2): 73-82. DOI: 10.1127/1432-8364/2010/0041.

Genuer, R., J. M. Poggi and C. Tuleau-Malot, 2010. Variable selection using random forests. *Pattern Recognition Letters*, **31**(14): 2225-2236. DOI: 10.1016/j.patrec.2010.03.014.

Gerke, M., 2014. Use of the stair vision library within the ISPRS 2D semantic labeling benchmark (Vaihingen). Technical report, ITC, University of Twente.

Gevers, T. and A. W. M. Smeulders, 1999. Color-based object recognition. *Pattern Recognition*, **32**(3): 453-464. DOI: Doi 10.1016/S0031-3203(98)00036-3.

Haralick, R. M., K. Shanmugam and I. Dinstein, 1973. Textural Features for Image Classification. *Ieee Transactions on Systems Man and Cybernetics*, **Smc3**(6): 610-621. DOI: Doi 10.1109/Tsmc.1973.4309314.

(1) http://rapidlasso.com/lastools/

Hariharan, B., J. Malik and D. Ramanan, 2012. Discriminative Decorrelation for Clustering and Classification. *Computer Vision - Eccv 2012, Pt Iv*, **7575**: 459-472.

Holzinger, A., 2015. Data Mining with Decision Trees: Theory and Applications. *Online Information Review*, **39**(3): 437-438. DOI: 10.1108/Oir-04-2015-0121.

Jain, A. K., N. K. Ratha and S. Lakshmanan, 1997. Object detection using Gabor filters. *Pattern Recognition*, **30**(2): 295-309. DOI: Doi 10.1016/S0031-3203(96)00068-4.

Kadir, T. and M. Brady, 2001. Saliency, scale and image description. *International Journal of Computer Vision*, **45**(2): 83-105. DOI: Doi 10.1023/A:1012460413855.

Klove, T., T. T. Lin, S. C. Tsai and W. G. Tzeng, 2010. Permutation Arrays Under the Chebyshev Distance. *Ieee Transactions on Information Theory*, **56**(6): 2611-2617. DOI: 10.1109/Tit.2010.2046212.

Kumar, S. and M. Hebert, 2003. Discriminative random fields: A discriminative framework for contextual interaction in classification. *Ninth Ieee International Conference on Computer Vision, Vols I and Ii, Proceedings*: 1150-1157. DOI: Doi 10.1109/Iccv.2003.1238478.

Kurani, A. S., D. H. Xu, J. Furst and D. S. Raicu, 2004. Co-occurrence matrices for volumetric data. *Proceedings of the Seventh IASTED International Conference on Computer Graphics and Imaging*: 447-452.

Li, Y. L., S. J. Wang, Q. Tian and X. Q. Ding, 2015. Feature representation for statistical-learning-based object detection: A review. *Pattern Recognition*, **48**(11): 3542-3559. DOI: 10.1016/j.patcog.2015.04.018.

Louppe, G., 2014. Understanding Random Forests: From Theory to Practice. *Eprint Arxiv*.

Ma, L., M. Li, T. Blaschke, X. Ma, D. Tiede, L. Cheng, Z. Chen and D. Chen, 2016. Object-Based Change Detection in Urban Areas: The Effects of Segmentation Strategy, Scale, and Feature Space on Unsupervised Methods. **8**(9).

Maclin, R. and D. Opitz, 1999. Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, **11**: 169-198.

Muller, K. R., S. Mika, G. Ratsch, K. Tsuda and B. Scholkopf, 2001. An introduction to kernel-based learning algorithms. *Ieee Transactions on Neural Networks*, **12**(2): 181-201. DOI: Doi 10.1109/72.914517.

Ojala, T., M. Pietikainen and T. Maenpaa, 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Ieee Transactions on Pattern Analysis and Machine Intelligence*, **24**(7): 971-987. DOI: Doi 10.1109/Tpami.2002.1017623.

Paisitkriangkrai, S., J. Sherrah, P. Janney and A. V. D. Hengel, 2015. Effective Semantic Pixel labelling with Convolutional Networks and Conditional Random Fields. *2015 Ieee Conference on Computer Vision and Pattern Recognition Workshops (Cvprw)*.

Palm, C., 2004. Color texture classification by integrative Co-occurrence matrices. *Pattern Recognition*, **37**(5): 965-976. DOI: 10.1016/j.patcog.2003.09.010.

Pauly, M., R. Keiser and M. Gross, 2003. Multi-scale feature extraction on point-sampled surfaces. *Computer Graphics Forum*, **22**(3): 281-289. DOI: Doi 10.1111/1467-8659.00675.

Qian, Y., X. W. Gao, M. J. Loomes, R. A. Comley, B. Barn, R. Hui and Z. Tian, 2011. Content-based petrieval of 3D medical images. *Iaria*: 303.

Qin, R., J. Tian and P. Reinartz, 2016. 3D change detection - Approaches and applications. *Isprs Journal of Photogrammetry and Remote Sensing*, **122**: 41-56. DOI: 10.1016/j.iprsjprs.2016.09.013.

Rokach, L., 2010. Ensemble-based classifiers. *Artificial Intelligence Review*, **33**(1-2): 1-39. DOI: 10.1007/s10462-009-9124-7.

Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Benitez, S. and Breitkopf, U., 2012. The ISPRS benchmark on urban object classification and 3D building reconstruction. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Melbourne, Australia, Vol. I-3, pp. 293–298.

Rouse, J. W., R. H. Haas, J. A. Schell and D. W. Deering, 1973. Monitoring Vegetation Systems in the Great Plains with ERTS. *Nasa Special Publication*, **351**: 309.

Wang, H., F. P. Nie, H. Huang and C. Ding, 2013. Heterogeneous Visual Features Fusion via Sparse Multimodal Machine. *2013 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr)*: 3097-3102. DOI: 10.1109/Cvpr.2013.398.

Weinmann, M. and M. Weinmann, 2018. Geospatial Computer Vision Based on Multi-Modal Data-How Valuable Is Shape Information for the Extraction of Semantic Information? *Remote Sensing*, **10**(1). DOI: Artn 2 10.3390/Rs10010002.

West, K. F., B. N. Webb, J. R. Lersch, S. Pothier, J. M. Triscari and A. E. Iverson, 2004. Context-driven automated target detection in 3-D data. *Automatic Target Recognition Xiv*, **5426**: 133-143. DOI: 10.1117/12.542536.