Hyperspectral Images Classification With Gabor Filtering and Convolutional Neural Network

Yushi Chen, Lin Zhu, Pedram Ghamisi, Xiuping Jia, Guoyu Li, and Liang Tang

Abstract—**Recently, the capability of deep learning-based approaches, especially deep convolutional neural networks (CNNs), has been investigated for hyperspectral remote sensing feature extraction (FE) and classification. Due to the large number of learnable parameters in convolutional filters, lots of training samples are needed in deep CNNs to avoid the overfitting problem. On the other hand, Gabor filtering can effectively extract spatial information including edges and textures, which may reduce the FE burden of the CNNs. In this letter, in order to make the most of deep CNN and Gabor filtering, a new strategy, which combines Gabor filters with convolutional filters, is proposed for hyperspectral image classification to mitigate the problem of overfitting. The obtained results reveal that the proposed model provides competitive results in terms of classification accuracy, especially when only a limited number of training samples are available.**

Index Terms—**Convolutional neural network (CNN), deep learning, feature extraction (FE), Gabor filtering, hyperspectral images (HSIs).**

I. INTRODUCTION

HYPERSPECTRAL images (HSIs) contain rich spectral–spatial information, which is found to be useful for a wide range of applications including agriculture, mineralogy, surveillance, physics, astronomy, chemical imaging, and environmental sciences [1]. Among the aforementioned applications, classification is a fundamental technique which has a greater influence on final performance. Due to the complexity of spectral and spatial structures, high dimensionality, and limited training samples, the classification of HSI still remains a challenging task [2].

Hyperspectral sensors provide rich spectral information, which can be used to differentiate classes with ever slightly different spectral characteristics. Therefore, traditional classification methods were focused on the use of spectral information alone (e.g., k-nearest neighbors, maximum likelihood, logistic regression, neural network, and support vector machine (SVM) [3]). Among those methods, kernel SVMs have demonstrated acceptable performances.

On the other hand, with respect to the recent developments in imaging spectrometry, it is now possible to find hyperspectral data sets with fine spatial resolution, which enables the analysis of small spatial structures on the surface of the earth [4]. To this end, it has been shown that the joint use of spectral and spatial information for HSI classification significantly improves classification accuracy values compared to the individual use of either spectral or spatial information. For example, in [5], it was shown that spatial information extracted by mathematical morphology and segmentation approaches has a significant influence on the classification performance.

Among the existing spatial information extraction techniques, Gabor filtering has attracted a lot of attention due to its capability to provide discriminative and informative features [6]. Compared with other filtering approaches, Gabor filtering shows its advantages in spatial information extraction including edges and textures [7].

In recent years, deep models have shown their capability in remote sensing research areas including optical remote sensing [8], synthetic aperture radar [9], and hyperspectral remote sensing [10]. In 2017, a timely survey letter overviewed the deep learning methods for remote sensing data processing [11].

In 2014, a deep learning-based method was proposed for the first time in the remote sensing community to deal with HSI classification [12]. In [12], stacked autoencoders were used to extract the hierarchical and robust features of HSI, which were useful for the classification step. Since then, many deep learning-based methods have been proposed, especially by considering convolutional neural networks (CNN). In [13], a deep CNN with five layers was employed to extract the spectral features of HSIs, leading to a promising classification performance. Furthermore, in [14], a framework including principal component analysis (PCA), deep CNN, and logistic regression was proposed. Very recently, a method based on the combination of morphological profiles and CNN was introduced to extract the features of HSI [15].

Deep learning methods, including CNNs, have a powerful feature extraction (FE) capability when sufficient training samples are available. Unfortunately, the lack of sufficient training samples is a common bottleneck in HSI classification. If the training samples are not sufficient, CNNs are often overtrained, which downgrades the quality of the classification maps and classification accuracies on the test samples. On the
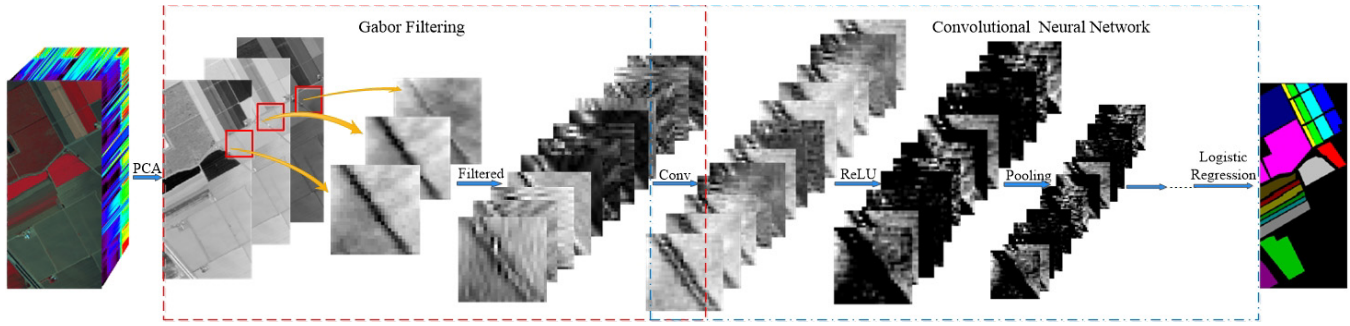
Fig. 1.　Framework of Gabor filtering and CNN for HSI classification.

contrary, Gabor filtering is an unsupervised technique for FE which is able to mitigate the overfitting problem in CNN-based HSI classification. In this letter, we utilize Gabor filtering to lessen the workload of CNNs and reduce the demand for a large number of training samples.

Furthermore, frequency- and orientation-based FEs in Gabor filtering are similar to those in the human visual system. On the other hand, CNN is a kind of neural network largely inspired by neuroscience. The inherent relationship between CNN and Gabor filtering makes the combination of the two techniques easier compared to the combination of CNN and other methods.

The main contributions of this letter are listed as follows.
1) A new classification framework is proposed for HSI, which is based on the combination of Gabor filtering and deep CNN. The proposed method works well under the condition of limited training samples compared with other traditional methods including CNN-based ones.
2) To boost the performance of the proposed method, advanced techniques including dropout, rectified linear unit (ReLU), and batch normalization (BN) are used and tested for HSI classification.

II. PROPOSED METHOD

A. Framework of the Proposed Method

The framework of the proposed method is shown in Fig. 1. The proposed methods used PCA on the spectral domain to reduce the dimensionality and Gabor filtering to extract the spatial features of HSIs. After that, the extracted features are fed into the CNN with BN and dropout to extract robust and discriminant features for the subsequent step. At the end of the framework, softmax is employed to produce the final classification map. The proposed framework is elaborated in detail in the following subsections.

B. Gabor Filtering

Gabor filtering effectively captures frequency and orientation representations in the spatial domain. In general, a Gabor filter consists of real and imaginary parts, which represents orthogonal directions. Therefore, a set of Gabor filters with different frequencies and directions are beneficial to precisely extract informative features. Gabor filter is defined by a Gaussian kernel function whose parameters f and θ represent the frequency and directions, respectively.

Supposing an HSI has n bands, the first step is to reduce its dimensions from n to $r (r \ll n)$ using PCA. Then, a set of fixed Gabor filters are applied to extract the edge and texture features from the images extracted by the PCA. These spatial

features can be further refined using deep networks at the next stage.

C. Convolutional Neural Network-Based Feature Extraction

The main three building blocks of CNNs are a convolution layer, a nonlinear transformation, and pooling. By stacking several convolution layers with nonlinear operation and several pooling layers, a deep CNN can be established. CNNs, by taking advantage of local connections and shared weights, tend to provide an appropriate performance.

A convolutional layer can be defined as follows:

$$x_j^l = f\left(\sum_{i=1}^{M} x_i^{l-1} * k_{ij}^l + b_j^l\right) \qquad (1)$$

where matrix x_i^{l-1} is the ith feature map of the previous $(l-1)$th layer, x_j^l is the jth feature map of the current (l)th layer, and M is the number of input feature maps. k_{ij}^l and b_j^l are randomly initialized and set to zero. Then, they are fine-tuned through backpropagation. $f(.)$ is a nonlinear function, and $*$ is the convolution operation.

Pooling operation offers invariance by reducing the resolution of the feature maps. The neuron in the pooling layer combines a small $N \times N$ (e.g., $N = 2$) patch of the convolution layer. The most common pooling operation is max pooling.

In order to incorporate the spatial information, $m \times m$ neighborhoods of the pixel to be classified are fed into the designed CNN model.

The success of CNNs mostly relies on the fact that the networks hierarchically learn the context invariant features, which are particularly useful for image classification.

D. Dropout, ReLU, and Batch Normalization

Due to the high dimensionality of HSI and limited available training samples, overfitting is a serious problem one may face. In order to handle the issue, a recently introduced method named dropout is used [16].

ReLU accepts the output of a neuron if it is positive, while it returns 0 if the output is negative. ReLU is a widely used nonlinear operator and has the advantages including sparse activation, efficient gradient propagation, and low computation load.

To further enrich the performance of the networks, an advanced technique named BN is also adopted in this letter [17]. BN is a practical tool in training deep neural networks for the following reasons: first, it can alleviate the problem caused by improper network initialization. Second, it can efficiently speed up the training procedure by preventing "gradient vanishing."
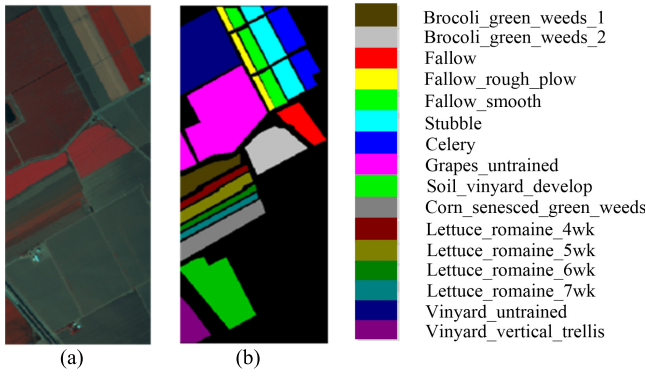
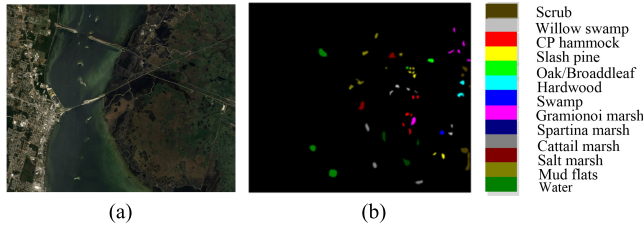Fig. 2. Salinas data set. (a) False-color map. (b) Ground-truth map.

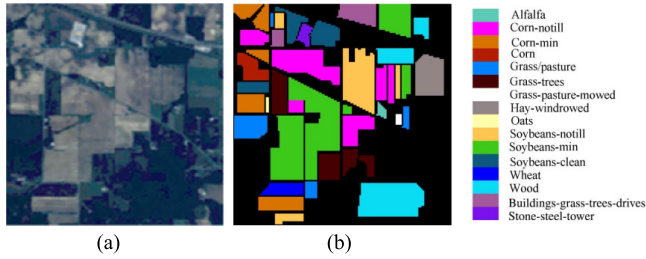Fig. 3. KSC data set. (a) False-color map. (b) Ground-truth map.

Fig. 4. Indian Pines data set. (a) False-color map. (b) Ground-truth map.

III. EXPERIMENTAL RESULTS

A. Data Description

In the experiments, three widely used hyperspectral data sets with different environmental settings were adopted to validate the performance of the proposed method. They have been captured over Salinas Valley (Salinas) in CA, USA, a site over the Kennedy Space Center (KSC) in FL, USA, and a mixed vegetation site in Northwest Indiana (Indian Pines). The false-color composite images and the corresponding ground-truth maps are demonstrated in Figs. 2–4.

B. Architecture of the Proposed Method

In general, Gabor filtering is regarded as a multidirectional and multiscale filter. In this letter, however, multidirectional and single-scale Gabor filters have been investigated in order to reduce the computational complexity and maintain robust feature representation capabilities simultaneously. Here, a grid search was used, which is a widely known method to find super-parameters and select the proper Gabor filters. The frequencies, which have been tested, are 0.1, 0.2, and 0.4 Hz. The four orientations ($\theta = 0$, $(\pi/4)$, $(\pi/2)$, $(3\pi/4)$) and eight orientations ($\theta = 0$, $(\pi/8)$, $(\pi/4)$, $(3\pi/8)$, $(\pi/2)$, $(5\pi/8)$, $(3\pi/4)$, $(7\pi/8)$) were examined for each frequency, and the best combination of the parameters is selected for those given parameters. We adopted one fixed frequency 0.2 Hz and four orientations to produce the

TABLE I
ARCHITECTURE OF THE DEEP CNNs

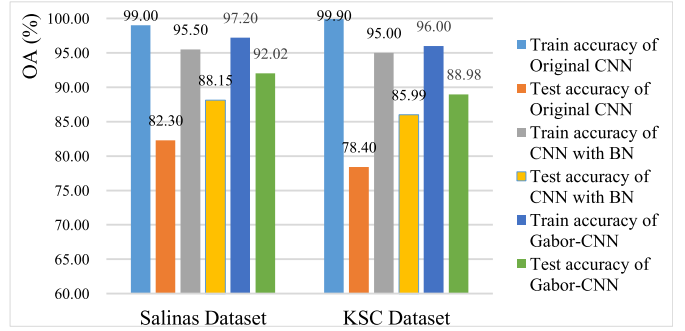| No.	Convolution	BN	ReLU	Pooling	Stride	Dropout
1	1×1×32	YES	YES	2×2	2	NO
2	5×5×48	YES	YES	2×2	2	50%
3	4×4×64	NO	YES	2×2	2	50%

Fig. 5. Train and test accuracies of different methods on Salinas (200 training samples) and KSC data sets (100 training samples).

Gabor filters with the size of $k \times k$ (e.g., $k = 3$). We performed Gabor filtering on r (e.g., $r = 3$) principal components of the PCA to get the spatial features. (The number of total extracted spatial features is $4r$.) The extracted $4r$ (e.g., 12) features are fed into the CNN in order to obtain more accurate classification maps.

In order to model spatial information, $m \times m$ (e.g., $m = 27$) neighborhood pixels were fed into the well-designed CNN. The generated architecture of the CNNs for each data set is shown in Table I. It contains three convolutional layers, three ReLU layers, and three pooling layers. The stride value was set to 2, and the dropout ratio was set to 0.5. In the end, all the nodes are equipped with fully connected layers followed by a softmax classifier to produce the corresponding labels. The training epochs are 160, 120, and 120 for Salinas, KSC, and Indian Pines data sets, respectively.

In these experiments, the classification accuracy is evaluated using overall accuracy (OA), average accuracy (AA), and Kappa coefficient (K). The proposed Gabor-CNN, which is compared with SVM with the radial basis function kernel (RBF-SVM), SVM with extended morphological profiles (EMP-SVM), original CNN, CNN with PCA (PCA-CNN), and EMP-CNN [14], [15]. In SVM-based methods, we searched the best parameters in the way of exponentially growing sequences of C and γ ($C = 10^{-4}, 10^{-3}, \ldots, 10^3$, $\gamma = 10^{-4}, 10^{-3}, \ldots 10^3$).

The training and test samples were randomly chosen among the whole samples. The experiments have run ten times with different initial random training samples. In order to compare the performance of the Gabor-CNN with the original CNN architecture, different numbers of training samples were tested.

C. Experimental Results

From Fig. 5, it can be seen that the original CNN faced a serious overfitting problem. For example, the train accuracy is 99.9%, but the test accuracy is descended to 78.4% on KSC data set. Gabor-CNN improves the test accuracies and mitigates the overfitting problem simultaneously. As an

TABLE II
TESTING DATA CLASSIFICATION RESULTS (VALUES ± STANDARD DEVIATION) ON SALINAS DATA SET

Methods \\ N		RBF-SVM	EMP-SVM	EMP-CNN	CNN	PCA-CNN	Gabor-CNN
800	OA(%)	85.69±1.03	88.24±0.85	**99.56±0.25**	98.88±0.62	99.12±0.18	99.53±0.14
	AA(%)	89.75±0.16	91.02±0.98	**99.78±0.54**	97.80±0.44	99.15±0.38	99.54±0.63
	K×100	89.38±1.20	86.89±0.95	**99.51±0.35**	98.41±0.19	99.01±0.13	99.47±0.16
400	OA(%)	85.74±0.46	88.69±1.57	95.78±0.19	96.33±0.29	96.67±0.58	**98.50±0.25**
	AA(%)	89.76±0.68	92.54±1.15	90.87±0.14	96.91±0.19	96.12±0.49	**98.06±0.09**
	K×100	84.07±0.54	87.42±1.70	95.27±0.21	96.06±0.63	96.28±0.57	**98.33±0.16**
200	OA(%)	83.09±1.08	85.90±1.26	87.04±0.16	88.15±0.24	89.26±0.48	**92.02±0.24**
	AA(%)	85.46±2.06	82.53±1.38	74.02±0.36	77.76±0.82	76.47±0.43	**88.95±0.29**
	K×100	81.07±1.19	84.02±1.46	85.43±0.19	86.05±0.13	87.88±0.52	**91.07±0.22**
100	OA(%)	76.09±1.34	79.90±1.54	80.04±0.16	82.92±0.56	88.02±0.28	**89.77±0.14**
	AA(%)	69.76±1.57	78.53±1.58	68.02±0.56	70.21±0.92	76.37±0.44	**87.90±0.20**
	K×100	73.08±1.45	77.92±1.06	78.43±0.48	80.79±0.43	87.28±0.92	**88.56±0.12**

TABLE III
TESTING DATA CLASSIFICATION RESULTS (VALUES ± STANDARD DEVIATION) ON KSC DATA SET

Methods \\ N		RBF-SVM	EMP-SVM	EMP-CNN	CNN	PCA-CNN	Gabor-CNN
300	OA(%)	81.41±1.20	97.19±0.37	98.42±0.12	97.98±0.34	98.12±0.12	**98.50±0.12**
	AA(%)	70.60±2.17	95.03±0.85	**97.76±0.03**	97.25±0.07	97.47±0.46	97.60±0.28
	K×100	79.27±1.32	96.88±0.41	98.24±0.13	97.67±0.53	98.05±0.13	**98.33±0.13**
200	OA(%)	76.13±0.24	90.59±0.25	96.94±0.13	95.63±0.26	96.02±0.42	**97.99 ±0.24**
	AA(%)	64.60±0.34	85.83±0.28	91.39±0.02	89.65±0.21	90.17±0.14	**96.14±0.40**
	K×100	73.52±1.30	89.41±1.39	96.60±0.14	94.95±1.13	95.64±0.49	**97.62±0.26**
100	OA(%)	70.84±1.38	85.23±0.92	85.58±0.39	85.99±1.45	86.53±0.42	**88.98±0.19**
	AA(%)	59.46±1.28	78.62±0.24	80.02±0.19	75.23±0.07	75.10±0.44	**80.58±0.19**
	K×100	67.51±1.58	83.54±1.87	84.17±0.37	84.43±0.17	84.98±0.51	**87.55±0.21**
50	OA(%)	67.74±1.18	71.49±0.42	75.37±0.29	77.79±1.15	78.57±0.22	**82.72±0.29**
	AA(%)	54.46±1.03	57.13±0.67	62.08±0.16	57.13±0.23	68.10±0.47	**74.77±0.43**
	K×100	63.88±1.23	68.20±1.42	72.50±0.37	68.20±0.47	74.58±0.31	**80.75±0.15**

TABLE IV
TESTING DATA CLASSIFICATION RESULTS (VALUES ± STANDARD DEVIATION) ON INDIAN PINES DATA SET

Methods \\ N		RBF-SVM	EMP-SVM	EMP-CNN	CNN	PCA-CNN	Gabor-CNN
400	OA(%)	62.90±1.03	76.62±0.85	**96.06±0.85**	93.80±0.42	93.76±0.48	95.19±0.13
	AA(%)	55.72±0.16	73.92±0.98	**84.28±0.74**	79.80±0.47	79.15±0.84	82.54±0.43
	K×100	57.38±1.20	73.36±0.95	**95.41±0.65**	92.71±0.16	92.61±0.34	93.67±0.76
300	OA(%)	59.24±0.46	73.19±1.57	**95.78±0.19**	91.30±0.29	91.67±0.28	92.30±0.25
	AA(%)	49.76±0.68	68.54±1.15	**90.87±0.14**	73.91±0.19	76.12±0.49	78.06±0.09
	K×100	53.27±0.54	69.11±1.70	**95.27±0.21**	89.66±0.63	89.08±0.17	90.23±0.46
200	OA(%)	58.01±1.08	69.34±1.06	86.48±0.16	86.47±0.54	87.26±0.28	**89.02±0.24**
	AA(%)	50.56±2.06	52.63±1.28	68.16±0.26	60.41±0.42	68.47±0.44	**70.95±0.29**
	K×100	52.07±1.19	64.56±0.56	84.23±0.26	84.12±0.19	83.08±0.32	**86.07±0.22**
100	OA(%)	51.90±0.64	58.23±0.54	75.84±0.16	77.42±0.56	78.86±0.28	**79.18±0.24**
	AA(%)	40.97±0.57	47.73±0.78	51.72±0.56	60.21±0.92	62.97±0.44	**62.20±0.42**
	K×100	45.08±1.45	50.52±1.06	71.43±0.48	73.79±0.43	75.08±0.92	**75.56±0.26**

example, Fig. 6 shows the classification maps on the Salinas data set. The detailed results were given in Tables II–IV. The proposed network has a superior performance in FE and classification. In this context, the Gabor-CNN achieves higher classification accuracy than that of CNN when the samples are reduced. Furthermore, the proposed method has also obtained better classification results compared with those of other studied methods.

In Table II, one can see that the OA of Gabor-CNN is 99.53% when the number of training samples is 800. This accuracy is higher than the one obtained by PCA-CNN. Moreover, our approach outperforms classical EMP-SVM by 11.29%, 8.52%, and 0.1258 in terms of OA,

AA and K, respectively. However, the accuracy is slightly lower than the one obtained by EMP-CNN. When the numbers of training samples are decreased by half, the performance of the CNNs is downgraded due to the availability of insufficient training samples. Under this condition, Gabor-CNN can perform better than CNNs. As shown in Table II, when the number of training samples is reduced, the proposed method demonstrates a remarkable performance compared to the CNN, EMP-CNN, EPM-SVM, and RBF-SVM.

The results in Table II demonstrate that Gabor filtering has the ability to alleviate the dependence of CNN on the number of training samples. In other words, Gabor-CNN mitigates the overfitting problem of original CNNs.
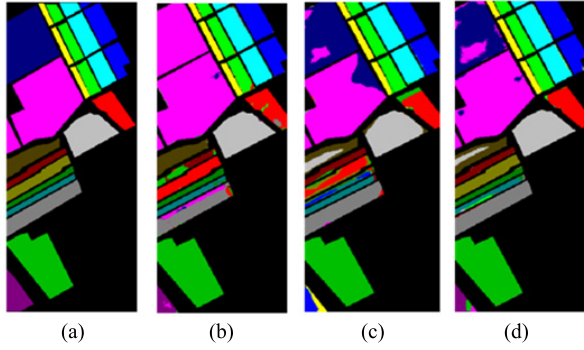
Fig. 6. (a) Ground truth of Salinas data set. The classification maps using (b) EMP-SVM, (c) PCA-CNN, and (d) Gabor-CNN.

TABLE V
RUNNING TIME OF THE CNN-BASED METHODS

Method	Running time (sec.)	Salinas	KSC	Indian Pines
Gabor-CNN	Training	49.26	26.78	50.26
	Test	2.46	1.76	2.85
PCA-CNN	Training	45.69	25.02	39.96
	Test	2.44	1.12	1.94
CNN	Training	34.82	15.56	30.52
	Test	2.15	1.04	1.56

Table III illustrates the classification results obtained by different approaches. For the KSC data set, we set the training samples as 300, 200, 100, and 50 for four different experiments. As can be seen, the OA of Gabor-CNN is 82.72 when the training samples are only 50. Furthermore, the OA of Gabor-CNN and PCA-CNN decreases by 9.5% and 11.6%, respectively, when the numbers of training samples are dropped from 300 to 100.

From Table IV, one can see that the comparison method, EMP-CNN, achieved good performance when the training samples are relatively large (e.g., 400 and 300). However, if the training samples are limited, the proposed Gabor-CNN outperforms all other methods in terms of OA, AA, and K.

From the obtained results detailed in Tables II–IV, one can find out that the Gabor-CNN method achieves a superior performance when the number of training samples is gradually reduced. The proposed method still maintains the highest OA, AA, and K compared to all other methods investigated in this letter. Furthermore, Gabor-CNNs have higher F scores than those of PCA-CNN for all the data sets tested. Taking all the aforementioned factors into consideration, we can conclude that the proposed Gabor-CNN method has a superior advantage in deep FE and classification of HSI, especially when only a limited number of training samples is available.

The running time of the CNN-based methods on all three data sets are shown in Table V. The experiments were run on a 3.2-GHz CPU with a GTX 1060 GPU card, where the numbers of training samples are set to100, 50, and 100 for the Salinas, KSC, and Indian Pines data sets, respectively. Furthermore, the running times (including training and test) of the complete grid search are 298.06, 162.45, 304.45 s on the Salinas, KSC, and Indian Pines data sets, respectively. In general, the deep CNN-based methods demand long processing time, but they are fast for the testing stage. However, deep learning algorithms can be implemented into a parallel version.

IV. CONCLUSION

In this letter, a new strategy, which combines the Gabor filtering and the CNN, was proposed for HSI classification. CNN is a powerful FE method, but it suffers from the problem of overfitting when the number of training samples is insufficient. In order to alleviate the overfitting problem of CNN-based methods, Gabor filtering was used to effectively extract spatial information including edges and textures.

The results obtained by the proposed method on Salinas, KSC, and Indian Pines data sets demonstrate that Gabor-CNN outperforms other studied methods when the number of training samples is limited. Thanks to the features extracted by the Gabor filters, CNNs can extract features and update weights effectively even if limited training samples are available.

REFERENCES

[1] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.

[2] L. Gómez-Chova, D. Tuia, G. Moser, and G. Camps-Valls, "Multimodal classification of remote sensing images: A review and future directions," *Proc. IEEE*, vol. 103, no. 9, pp. 1560–1584, Sep. 2015.

[3] P. Ghamisi, J. Plaza, Y. Chen, J. Li, and A. J. Plaza, "Advanced spectral classifiers for hyperspectral images: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 1, pp. 8–32, Mar. 2017.

[4] J. A. Benediktsson and P. Ghamisi, *Spectral-Spatial Classification of Hyperspectral Remote Sensing Images*. Boston, MA, USA: Artech House, 2015.

[5] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveinsson, "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 11, pp. 3804–3814, Nov. 2008.

[6] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 55–73, Jan. 1990.

[7] T. Randen and J. H. Husøy, "Filtering for texture classification: A comparative study," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 4, pp. 291–310, Apr. 1999.

[8] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[9] S. Chen, H. Wang, F. Xu, and Y.-Q. Jin, "Target classification using the deep convolutional networks for SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4806–4817, Aug. 2016.

[10] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.

[11] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 22–40, Jun. 2016.

[12] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.

[13] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, Jan. 2015, Art. no. 258619.

[14] J. Yue, W. Zhao, S. Mao, and H. Liu, "Spectral–spatial classification of hyperspectral images using deep convolutional neural networks," *Remote Sens. Lett.*, vol. 6, no. 6, pp. 468–477, May 2015.

[15] E. Aptoula, M. C. Ozdemir, and B. Yanikoglu, "Deep learning with attribute profiles for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1970–1974, Dec. 2016.

[16] G. Hinton *et al.*, "Improving neural networks by preventing co-adaptation of feature detectors," *Comput. Sci.*, vol. 3, no. 4, pp. 212–223, Apr. 2012.

[17] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," Unpublished paper, 2015. [Online]. Available: https://arxiv.org/abs/1502.03167