A Hybrid Approach for Remote Sensed Hyperspectral Images Classification

S. L. J. L. Tinoco, G. Cámara-Chávez, D. Menotti
UFOP - Federal University of Ouro Preto
Computing Department
Ouro Preto, MG, Brazil
Email: sjtinoco@yahoo.com.br,{gcamarac,menottid}@gmail.com

Abstract—**This paper presents a hybrid classification approach combining the use of supervised classification and unsupervised clustering algorithms. The main idea is to reduce the training set by selecting the most appropriated samples for classification by means of K-nearest neighbor (KNN) algorithm. Indeed, for each class the resulting center clusters from Kmeans are chosen as those samples. Experiments are carried out using two well-known databases: Indian Pines, acquired by AVIRIS sensor; and Pavia University, acquired by ROSIS sensor. Results show the efficiency of our proposed approach which significantly reduces the time required in the classification step while the effectiveness/accuracy is kept close to the ones of the original KNN.**

I. Introduction

The emergence of remote sensed hyperspectral images has brought some challenges to the task of data interpretation. Among them we may mention the modeling of high dimensional data and their parameter estimation. In multispectral data (dozens of spectrals), Gaussian distribution model has been used for these purposes [8]. However, when dealing with hyperspectral imaging a large number of training samples for each class is required in order to estimate the terms of the large covariance matrices, for example. It should be noted also that a unimodal Gaussian description is not enough to handle multimodal data class [10]. In order to circumvent the above problems, the use of non-parametric algorithms such as the k-nearest neighbors (KNN) can be a good choice, since it has the advantage of not requiring estimated density function for each class [10]. Despite its simplicity, the KNN has been widely used [1], [12], [11], [18], having a high degree of accuracy, clarity of its working/rules and it is easy to implement.

The K-Nearest Neighbor (KNN) is one of the most simple and intuitive algorithms to supervised classification. It is assumed that nearest samples are in the same class. This notion is used for the classification task and the KNN works as follows. For each unclassified pattern (*testing set*), one seek for the closest known class patterns (*training Set*) in the feature space, *i.e.*, the nearest neighbors. And it uses the class of these classified samples to selecting by majority the class of the unclassified pattern.

The KNN classifier is the one in which learning is based on analogy. The training set is composed by patterns represented by a n-dimensional vectors. Each pattern of this group can be seen as a point in a n-dimensional space. In order to determine the class of an pattern which does not belong to the training set, the classifier KNN chooses the patterns of the training set that are closest to this unknown pattern, *i.e.*, having the greatest similarity, usually the smallest distance. Computational cost can be high due to the number of comparisons to be made [2], since the similarity/distance from the unclassified pattern to the all training set has to be computed. That is, a large number of spectral distances should be evaluated for each pixel which requires a high computational load, especially when the number of spectral bands and/or the number of training samples is large. This is why the KNN has been primarily limited to the classification of multispectral or hyperspectral data. It is used only after features reduction has been achieved [10].

Thus, an approach which may take advantage of KNN reducing its computational cost can be useful and appropriate to classify remote sensed hyperspectral images. With this in mind, the connection between an unsupervised classification algorithm as the Kmeans and nonparametric KNN is proposed in this work. In order to reduce the computational load, we suggest to reduce the training set size by selecting the most appropriated samples. For each class, these samples are chosen as the resulting center clusters from Kmeans.

The remainder of this paper is organized as follows. Section II presents our approach for remote sensed hyperspectral images classification. Section III describes the experiments performed using two well-know database (Indian Pines, acquired by AVIRIS sensor [13]; and Pavia University, acquired by ROSIS sensor [3].) in order to validate the proposed approach. Finally, conclusions are pointed out in Section IV.

II. The proposed approach

The proposed approach aims to obtain a reduced training set such that the KNN classification algorithm run faster than in its original way. Moreover, we expect that the instances chosen for each class, which are cluster centers of Kmeans, could keep the classification effectiveness similar to the one when all training set is used. A flowchart of our proposed approach is shown in Fig. 1.

A. Kmeans

The Kmeans clustering algorithm is a partition that is characterized by dividing the dataset into disjoint subsets.

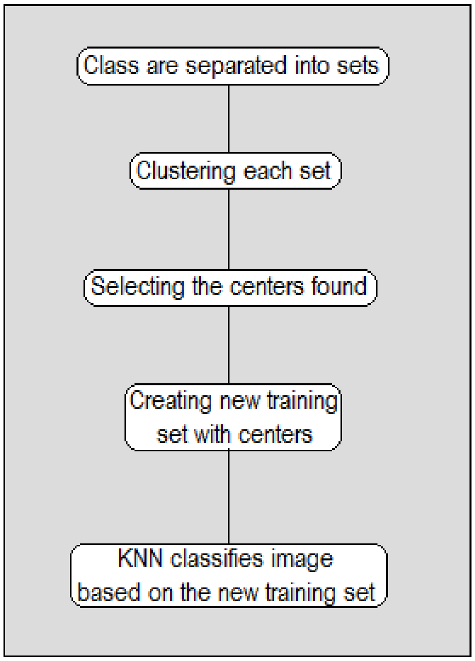

Class are separated into sets
Clustering each set
Selecting the centers found
Creating new training set with centers
KNN classifies image based on the new training set

Fig. 1: Scheme of the proposed approach

According to [19], the Kmeans does not require spatial information and has the great advantage of the computational time. It is based on distance, since its function is similarity in the distance, which seeks to minimize. The most popular clustering algorithm is the Kmeans [9] using Euclidean distance. The idea of the algorithm is to provide a classification according to the data itself, based on analysis and comparisons of their numerical values. Thus, the algorithm provides an automatic classification without the need for human supervision. Because of this feature, the Kmeans algorithm is considered as a data classification unsupervised. According to a pre-defined rule, this method uses values from the data itself as temporary estimates of the average of clusters Km, where Km is the number of clusters specified by the user (Fig. 2). Thus, the center of the initial cluster is formed for each case around the data next and then compared with the more distant points and the others formed clusters. For each class that has more than a point value of the new centroid is calculated by the mean of each attribute of all points belonging to this class. Thereafter, within a process of continuous updating and an iterative process are the final cluster centers.

The proposed strategy is the implementation of the clustering algorithm on each set of instances of the same class in the original image. Then from each cluster obtained, their centers will be selected and form the new training set for the KNN. This new training set is then formed by data representing best determined class, thus diminishing the effect of intra-class, and of noise in the set clustered (Fig. 2).

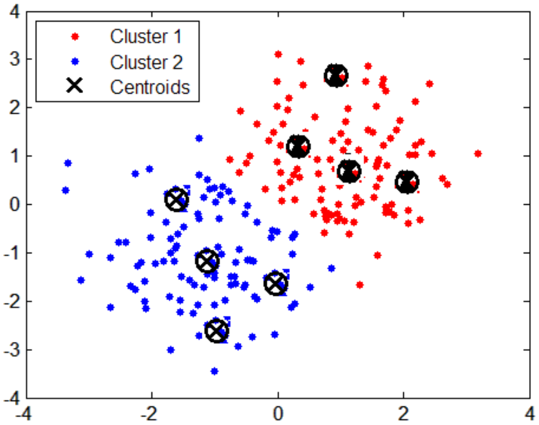


Fig. 2: Illustrating example of the use of Kmeans clustering algorithm for training set reduction

B. KNN

The K-Nearest Neighbor (KNN) [2] is one of the most simple and intuitive to supervised classification, it is assumed that within the close two instances of attributes are the same class. For each pattern with unknown class, look for the patterns of known class (Training Set) closest in feature space, the nearest neighbors, and uses the class closer to these standards for classification, choosing the class corresponding to majority. The KNN classifier is one in which learning is based on analogy. The training set consists of n-dimensional vector and each element of this set represents a point in n-dimensional space. To determine the class of an element that does not belong to the training set, the KNN classifier seeks to K elements of the training set that are closest to this unknown element, that is, having the shortest distance. Let a general KNN rule be

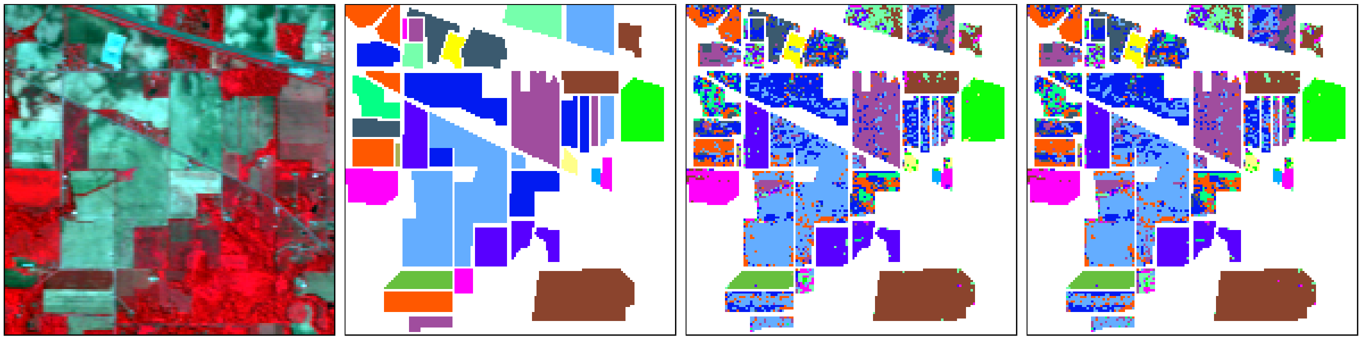
$$x \in \omega_i, \text{ if } m_i > m_j \text{ for all } j \neq i \tag{1}$$

where $m_i(x)$ is the membership that pixel vector x belongs to class i. For the basic KNN

$$m_i(x) = k_i \tag{2}$$

In this work, Euclidean distance is used as the spectral distance measure.

Computational cost can be high because the number of comparisons to be made. According to the literature, this algorithm is a good classifier, although simple, and has been widely used today. Theoretically, the KNN is optimum in terms of accuracy when the training set tends to infinity. On the other hand, satisfactory results are achieved with small K values (typically less than 10). But there is always a linear cost ($O(n)$ where n is the size of the training set) associated with the classification of each sample. In this regard, several studies have been proposed in order to reduce the training set to improve the classification of each time sample or training classifiers.


(a) Original

(b) Ground Truth

(c) Thematic Map for 3-NN

(d) Thematic Map for 5-NN and 60 clusters

Fig. 3: *Indian Pines* dataset, 200 bands, AVIRIS sensor

Thus, reduction of the training set of KNN is desired, however, must be preserved better represent the features that each class. The proposed approach provides for the KNN algorithm a set of reduced training, thus reducing its running time.

III. Experiments and Results

In this sections, we describe the experiments performed in order to validate the our proposed approach. Two well-know datasets are used in our experiments: Indian Pines, acquired by AVIRIS sensor [13]; and Pavia University, acquired by ROSIS sensor [3]. In the following we describe how these data are organized. In Sections III-A and III-B, two experiments are detailed, and finally, in Section III-C, an analysis is presented.

In order to determine the reliability of the constructed model with the data available, the N-fold cross validation scheme is employed, in which the dataset is divided into N subsets. Among these subsets, one is retained to be used as testing and the remaining $N-1$ subsets are used for training. The validation procedure is repeated N times until each subset is used exactly once as testing data, as illustrated in Fig. 4. In this way, the N average effectiveness of the classifier in testing is obtained.

The dataset division is performed as follows. The labeled pixels are divided into sets, in which each set represents a class. Then each class set is equally divided into five subsets ($N = 5$). The resulting subsets are grouped so that each contain $1/5$ of the labeled pixels of each class.

It is important to note that the number of classes in both images/datasets is quite unbalanced, *i.e.*, some few classes contain the majority of pixels while others have many few, as can be seen in column *Samples* in Tables II and V. Therefore, when applying the clustering algorithm and selecting the cluster centers found, it may happen that the classes with more elements are not well represented, since the number of centers is equal for all classes. This procedure can reduce the accuracy of classification.

In order to found a better had the representation for the new training set extracted from the cluster centers, we adopted the following strategy. Firstly, there is a quantity Q of elements for each class and the median M calculated between the values found. M is divided by the number of clusters Km, previously reported, resulting in R. Then the new number of clusters NKm to be used in each class is now the value for the quantity Q of the respective cluster divided by R.

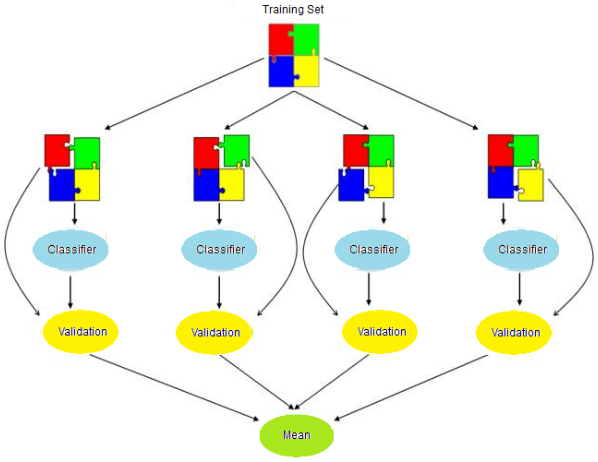
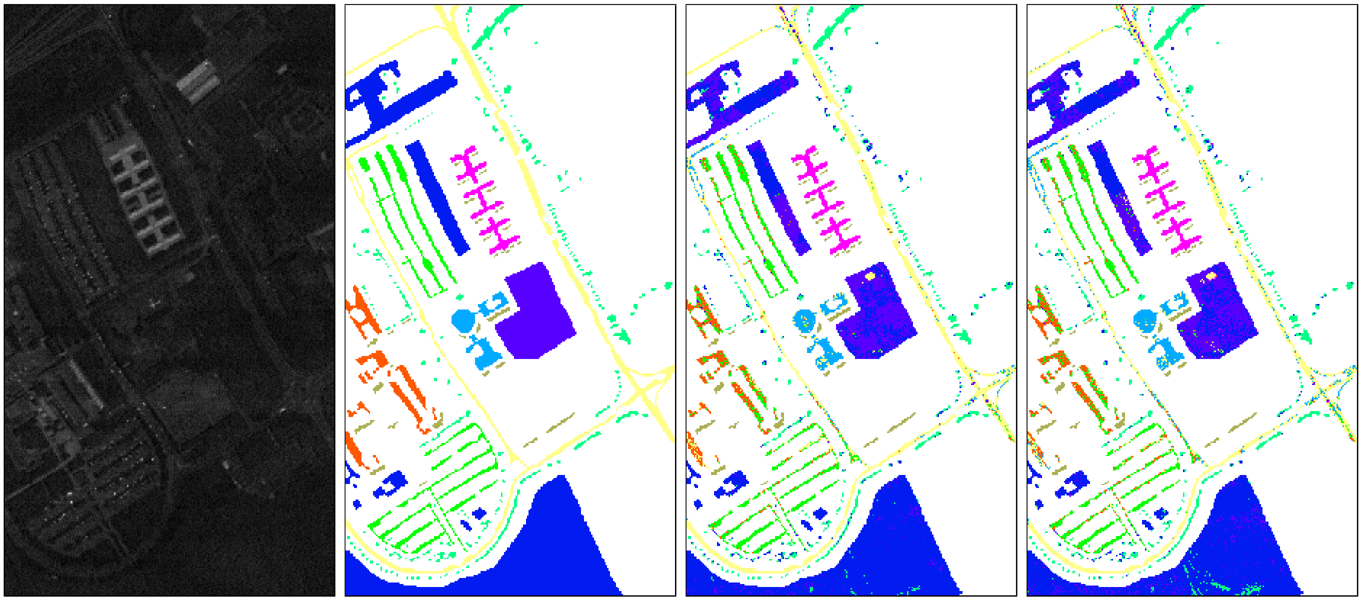


Fig. 4: N-folds cross validation scheme

A. *Experiment 1*

Experiments are performed using the *Indian Pines* datasets, acquired by AVIRIS airborne sensor data [13], which cover an area of agriculture and forest in northeastern Indiana, USA, $145 \times 145 \times 220$ pixels. Noise bands are removed, that is, the indexed from 104 to 108, from 150 to 163 and 220, remaining a total of 200 bands. This image presents sixteen classes or categories as can be seen in Fig.'s 3a and 3b.

This image is classified using KNN with full training set and the proposed approach, and the respective obtained thematic maps are shown in Fig.'s 3c and 3d. The obtained figures are shown in Table I. As we can observe, the accuracy of KNN is higher, but the time required for classification is greater than that of the proposed approach. In these experiment, we used



(a) Original

(b) Ground Truth

(c) Thematic Map for 3-NN

(d) Thematic Map for 1-NN and 60 clusters

Fig. 5: *Pavia University*, 103 bands, ROSIS sensor

all 200 image bands, which may influenced the low accuracy. Note that the time required for clustering is taken into account.

TABLE I: Overall Accuracy for *Indian Pines*

K	Km	Accuracy	Standard Deviation	Time
1	–	66.82%	5.27	11min 37s
3	–	66.96%	5.21	11min 52s
5	–	66.56%	5.22	11min 59s
1	5	60.64%	5.26	59s
1	20	65.91%	5.30	4min 25s
1	40	67.05%	5.28	7min 46s
1	60	66.86%	5.28	9min 55s
3	5	59.07%	5.12	59s
3	20	64.45%	5.23	4min 17s
3	40	66.64%	5.22	8min 3s
3	60	66.46%	5.20	9min 56s
5	5	56.26%	5.20	59s
5	20	64.02%	5.27	4min 23s
5	40	66.10%	5.24	7min 53s
5	60	66.89%	5.19	9min 56s

Table II and III shows the obtained accuracy for each class for our proposed approach and for the KNN using the full training set, respectively.

TABLE III: Accuracy (%) per class for *Indian Pines* dataset using KNN and the full training set

Class Number	Class	Samples	KNN = 1	KNN = 3	KNN = 5
1	Alfafa	54	64.81	64.81	61.11
2	Corn-notill	1434	46.79	49.86	47.28
3	Corn-mintill	834	44.84	44.00	42.56
4	Corn	234	47.86	36.75	36.75
5	Grass-pasture	497	77.26	74.64	73.84
6	Grass-trees	747	94.10	95.18	96.25
7	Grass-pasture-mowed	26	84.61	84.61	84.61
8	Hay-windrowed	489	96.11	97.13	97.34
9	Oats	20	60.00	60.00	55.00
10	Soybean-notill	968	67.25	66.63	68.59
11	Soybean-mintill	2468	61.38	62.96	63.53
12	Soybean-clean	614	44.46	41.53	37.45
13	Whea	212	96.69	97.16	97.64
14	Woods	1294	91.19	92.89	93.66
15	Buildings-Grass-Trees-Drives	380	55.52	46.84	42.89
16	Stone-Steel-Towers	95	86.31	84.21	86.31

B. *Experiment 2*

In order to verify the degree of generalization of our approach, tests are performed with a second training set. An image of the *University of Pavia*, Italy, acquired by the sensor ROSIS, 610 × 340 × 103 pixels is used [3]. These image presentes nine classes as can be seen in Fig.'s 5a and 5b.

The experiments are performed with all 103 bands image. The results with the KNN and proposed approach can be seen in Table IV. The proposed approach obtained an accuracy slightly lower than the KNN using the full training data, however its running time is quite smaller. Table V and VI shows the obtained accuracy for each class for our proposed approach and for the KNN using the full training set, respectively.

TABLE IV: Overall Accuracy for *Pavia University* dataset

K	Km	Accuracy	Standard Deviation	Time
1	–	80.45%	1.83	1h 37min 19s
3	–	81.34%	1.81	1h 37min 12s
5	–	81.16%	1.81	1h 38min 48s
1	5	73.80%	1.83	5min 47s
1	20	77.14%	1.82	8min 3s
1	40	78.42%	1.84	12min 42s
1	60	78.88%	1.85	22min
3	5	73.10%	1.82	5min 55s
3	20	76.67%	1.79	7min 58s
3	40	78.73%	1.82	12min 58s
3	60	78.41%	1.84	22min 39s
5	5	73.27%	1.81	5min 58s
5	20	76.19%	1.79	8min 9s
5	40	78.32%	1.81	13min 33s
5	60	77.97%	1.84	23min 4s

C. *Analysis*

Analyzing the results shown in Table I (*Indian Pines*-AVIRIS) and in Table IV (*Pavia University*-ROSIS), we can observe that the accuracies reached by the proposed approach

TABLE II: Accuracy (%) per class for *Indian Pines* dataset using our proposed approach

#	Class	Samples	K - KNN											
			1				3				5			
			Km - Kmeans											
			5	20	40	60	5	20	40	60	5	20	40	60
1	Alfafa	54	68.51	64.81	64.81	64.81	77.77	64.81	64.81	64.81	79.62	61.11	61.11	61.11
2	Corn-notill	1434	41.35	43.37	44.28	45.39	50.41	46.02	43.44	50.90	40.02	42.18	37.93	48.81
3	Corn-mintill	834	35.61	39.56	45.68	44.72	28.41	36.57	47.00	45.32	22.42	36.45	49.16	44.60
4	Corn	234	16.66	57.26	51.28	48.29	12.39	56.83	41.88	39.31	6.41	56.83	41.02	38.46
5	Grass-pasture	497	68.20	77.86	77.26	77.06	65.79	76.05	75.65	75.85	58.35	75.85	75.45	74.24
6	Grass-trees	747	90.76	90.89	94.37	94.10	91.29	87.81	95.18	95.18	91.03	86.74	96.11	96.11
7	Grass-pasture-mowed	26	88.46	84.61	84.61	84.61	88.46	84.61	84.61	84.61	88.46	84.61	84.61	84.61
8	Hay-windrowed	489	88.54	96.11	96.11	96.11	76.48	97.13	97.13	97.13	68.09	97.34	97.34	97.34
9	Oats	20	65.00	75.00	60.00	60.00	75.00	60.00	60.00	60.00	75.00	55.00	60.00	60.00
10	Soybean-notill	968	61.98	63.11	65.08	66.21	60.02	64.77	69.93	67.66	61.26	67.35	73.65	70.24
11	Soybean-mintill	2468	57.57	63.69	62.76	62.56	57.53	62.56	61.46	60.61	58.63	63.77	60.94	60.29
12	Soybean-clean	614	30.94	37.78	48.04	45.27	16.28	20.68	45.92	41.53	15.14	18.56	43.97	38.59
13	Whea	212	92.45	96.69	96.69	96.69	92.45	97.64	97.16	97.16	91.98	98.11	97.64	97.64
14	Woods	1294	94.66	90.41	91.03	91.19	95.05	90.18	91.49	92.89	94.82	89.87	92.04	93.66
15	Buildings-Grass-Trees-Drives	380	23.68	63.15	57.89	55.78	9.47	59.21	50.26	47.63	3.15	55.00	45.26	43.42
16	Stone-Steel-Towers	95	92.63	86.31	86.31	86.31	89.47	86.31	84.21	84.21	86.31	86.31	86.31	86.31

TABLE V: Accuracy (%) per class for *Pavia University* dataset using our proposed approach

#	Class	Samples	K - KNN											
			1				3				5			
			Km - Kmeans											
			5	20	40	60	5	20	40	60	5	20	40	60
1	Asphalt	6631	82.14	84.24	83.77	80.28	81.70	84.57	86.03	79.52	79.70	83.90	84.91	77.52
2	Meadows	18649	74.21	77.17	78.04	79.03	78.26	79.87	80.10	80.20	81.04	80.67	80.80	80.96
3	Gravel	2099	47.78	56.02	58.31	60.98	18.48	41.73	47.92	53.59	17.00	39.73	47.59	53.02
4	Trees	3064	85.93	88.54	87.20	87.92	80.87	87.43	86.68	87.27	75.84	84.72	85.73	86.39
5	Painted metal sheets	1345	94.72	98.81	99.33	99.70	97.84	98.73	99.25	99.55	98.21	98.66	99.40	99.55
6	Bare Soil	5029	54.38	58.38	80.45	67.01	45.09	50.32	60.31	61.42	38.65	44.85	55.55	57.10
7	Bitumen	1330	69.62	81.05	84.61	86.31	71.65	80.37	81.72	90.30	75.41	82.40	81.72	91.20
8	Self-Blocking Bricks	3682	74.76	76.80	77.97	78.35	78.76	77.48	80.25	79.41	82.53	79.19	81.39	80.66
9	Shadows	947	99.57	99.78	99.78	99.78	99.47	99.78	99.78	99.78	99.47	99.68	99.78	99.78

TABLE VI: Accuracy (%) per class for *Pavia University* dataset using KNN and the full training set

lass Number	Class	Samples	KNN = 1	KNN = 3	KNN = 5
1	Asphalt	6631	84.22	86.56	86.20
2	Meadows	18649	81.23	82.50	82.90
3	Gravel	2099	63.93	64.50	63.93
4	Trees	3064	85.24	84.10	83.45
5	Painted metal sheets	1345	99.55	99.40	99.47
6	Bare Soil	5029	68.36	66.09	64.40
7	Bitumen	1330	80.60	83.00	82.63
8	Self-Blocking Bricks	3682	79.22	81.74	81.55
9	Shadows	947	99.68	99.78	99.78

is very close to the ones reached by the KNN using the full training dataset. However, the proposed approach obtained run times much lower than the KNN when using the full training dataset.

Tables II and III (*Indian Pines*-AVIRIS) and in Tables V and VI (*Pavia University*-ROSIS) show the accuracy for each class. Depending on the purpose of classification, for example, identify only urban areas (represented by a class), the accuracy per class may be more important than overall accuracy. Also from this result, we can observe how classes with the largest number of elements are classified.

IV. Conclusions

In this paper, we presented a hybrid approach for remote sensed hyperspectral images classification, linking a clustering (Kmeans) and a supervised non-parametric classification (KNN) algorithms. From the experiments using two well-know databases (Indian Pines, acquired by AVIRIS sensor [13]; and Pavia University, acquired by ROSIS sensor [3]), we can observe that the obtained accuracy by the proposed approach is close to the ones obtained by the KNN with the full training data. Regarding the runtime, the proposed approach achieved much better results being up to ten times faster than KNN.

As future work, we plan to study other clustering algorithms such as: ISODATA [6], DBSCAN [4], DenClust [7], Xmeans [16], Optimum-path forest [15], [17], [14], etc. We also plan to study algorithms developed for sub-spaces clustering on high dimensional [5] such that, the KNN can process. In this way, we expect to decrease even more the KNN run time keeping the obtained accuracy close to the original values.

References

[1] E. Blanzieri and F. Melganin. Nearest neighbor classification of remote sensing images with the maximal margin principle. *IEEE Transactions on Geoscience and Remote Sensing*, 46(6):1804–1811, 2008.

[2] P. Cover, T. e Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 1967*, 13(1):21–27, 1967.

[3] U. del Pais Vasco. http://www.ehu.es/ccwintco/index.php/Hyperspectral-Remote-Sensing-Scenes. last visit on August, 25th, 2012.

[4] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Knowledge Discovery and Data Mining*, pages 226–231, 1998.

[5] G. Gan, C. Ma, and J. Wu. *Data Clustering: Theory, Algorithms, and Applications*. Society for Industrial and Applied Mathematics, 2007.

[6] D. J. Hall and G. B. Ball. Isodata : A novel method of data analysis and pattern classification. *Journal of Machine Learning Research*, 1965.

[7] A. Hinneburg and D. A. Keim. An efficient approach to clustering in large multimedia databased with noise. In *Knowledge Discovery and Data Mining*, pages 58–65, 1998.

[8] J. P. Hoffbeck and D. A. Landgrebe. Classification of remote sensing images having high spectral resolution. *Remote Sensing of Environment*, 57(3):119–126, 2009.

[9] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review,1999. *Computational Statistics and Data Analysis, 1999*, 31(3):264–323, 1999.

[10] X. Jia and J. A. Richards. Fast k-nn classification using the cluster-space approach. *IEEE Geoscience and Remote Sensing Letters*, 2(2):225–228, 2005.

[11] Y. Li and B. Cheng. An improved k-nearest neighbor algorithm and its application to high resolution remote sensing image classification. *In IEEE International Conference on GeoInformatics, 2009*, pages 1–4, 2009.

[12] D. O. Mcinerney and M. Nieuwenhuis. A comparative analysis of knn and decision tree methods for the irish national forest inventory. *International Journal of Remote Sensing*, 30(19):4937–4955, 2009.

[13] MultiSpec. https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html. last visit on August, 25th, 2012.

[14] J. P. Papa, A. X. Falco, V. H. C. de Albuquerque, and J. M. R. Tavares. Efficient supervised optimum-path forest classification for large datasets. *Pattern Recognition*, 45(1):512–520, 2012.

[15] J. P. Papa, A. X. Falco, and C. T. N. Suzuki. Supervised pattern classification based on optimum-path forest. *International Journal of Imaging Systems and Technology*, 19(2):120–131, 2009.

[16] D. Pelleg and A. W. Moore. X-means: Extending k-means with efficient estimation of the number of clusters. In *International Conference on Machine Learning (ICML)*, pages 727–734, 2000.

[17] L. M. Rocha, F. A. M. Cappabianco, and A. X. Falco. Data clustering as an optimum-path forest problem with applications in image analysis. *International Journal of Imaging Systems and Technology*, 19(2):50–68, 2009.

[18] L. Samaniego, A. Brdossy, and K. Schulz. Supervised classification of remotely sensed imagery using a modified k-nn technique. *IEEE Transactions on Geoscience and Remote Sensing*, 46(7):2112–2125, 2008.

[19] T. N. Tran, R. Wehrens, and L. M. C. Buydens. Clustering multispectral images: a tutorial. *Chemometrics and Intelligent Laboratory Systems, 2005*, 77(1):3–17, 2005.