RESEARCH ARTICLE OPEN ACCESS

Target Class Guided Compression in Feature Subspace

H. N. Meenakshi [1], P. Nagabushan [2]
Department of Studies in Computer Science
University of Mysore
Karnataka - India

ABSTRACT

Driven by the target class, a hybrid algorithm is developed in this paper to encode the high dimensional data which consists of several classes. The main characteristic of the proposed hybrid algorithm is to encode the target class so as to ensure its lossless recovery as required by the user and to encode the other classes with lossy of near-lossy technique. Before applying the encoding process, the high dimensional data is first projected on to the feature subspace obtained from the knowledge of the target class, which could consequently increase the density within the target class and the scatterness among the samples belonging to other unknown classes. This dual effect on the reduced feature space of the input is utilized to develop hybrid compression based on vector quantization technique. Lossless encoding is designed by generating the primary code vectors using the training samples of the target class based on K-means clustering and a lossy encoding of other samples is designed based on ISODATA by generating secondary code vectors. A primary codebook is so designed that the possibility of the distortion in the target class gets minimized. One face of the proposed hybrid compression ensures the lossless recovery of the target class while the other face provides a good compression ratio through lossy compression. The experimentation is conducted on AVIRIS and ROSIS remotely sensed standard bench mark data sets and the results are then compared with a compression carried out by vector quantization on the original space, JPEG and JPEG2000.

Keywords :— Target Class, Class of Interest-CoI, Feature subsetting, Primary code book, secondary codebook, distortion, encoder, decoder, K-means, ISODATA, bit rate.

I. INTRODUCTION

A remotely sensed hyperspectral data which covers a large geographical area consists of several classes and is also a rich source of information for various applications. However, at any given time an application would focus on just one class called the target class/Class of interest than on any other classes. For example, while monitoring the grazing area using the remotely sensed archive, it is necessary to use the supplied high quality vegetation class and it could be sufficient even when a degraded quality type of data is provided. A similar situation can be noticed in the online retailing websites which gather extremely large amounts of data every second and when let for a day or a year it then accumulates to several petabytes of data that would result in big data. The major reason behind the success of such online retailing is its value in terms of customer recommendations. The user will be advised to purchase based on the browsing history of the items being searched or purchased earlier. While this is a common occurrence today, Amazon was seen as being one of the first companies to put a focus on using its big data to give its customers a personalized feel and focused buying experience. The duration up to which the browsing history of each customer is stored in these websites varies from a few weeks to several months. In such a case, it could be advantageous to store the very recent history (the past few weeks) precisely whereas the very old history (a few months old) imprecisely, while correlating the browsing history. Another supporting example of the same kind is from the medical domain. For a specific case in medicine it is enough to maintain a superior quality of the image for just the region to be diagnosed while the rest of image may be encoded with lower quality.

The aforementioned examples considered from various domains signify the need for customizing the compression based on the user's class of interest-CoI. Therefore, in this paper a new compression technique is designed to compress the CoI so that it can be decoded with void loss. On the other hand, the samples that are not the member of CoI are encoded using a lossy technique which after decoding might results in a lossy or near lossy data. Such a selective hybrid compression strategy offers the following advantages. The lossless encoding of the CoI samples is expected to ensure the original quality of the decoded CoI as required by the user. On the other hand the lossy compression of the other samples would ensure the effective compression in terms of accomplishing the maximum compression ratio. While the user is not apprehensive about the quality of the other samples, the loss or nearly lossy counter CoI samples resulting from the decompression doesn't affect the processing of the data in an application. The suitable application of the planned CoI based compression is as shown in figure 1. The data distribution centre collects the data from the various resources and before distributing it to the user it could be compressed focusing the CoI as required by that particular user. After receiving the compressed data the user can decompress, extract and process the required class from among several classes present in it. Later, if the user prefers to store the processed data then it advisable to also apply the CoI based compression.

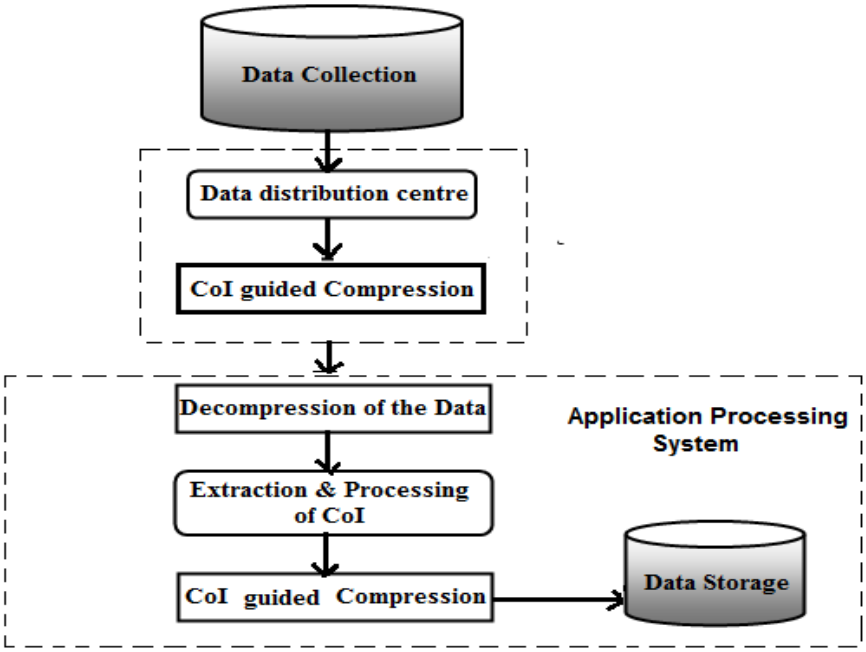

Fig1. Illustration of the feasible relevance of the proposed CoI based compression technique.

Suppose the data to be compressed is of a high dimensional type then, strategically it is feasible to achieve a maximum compression ratio. Conversely, the high dimensional property of the data is not favourable when an accurate extraction of the target class from among several classes is anticipated by the user from the decoded input space. Assuming that the entire high dimensional data is encoded in its original form without any pre-processing like dimensionality reduction, and if the same is decoded then it could results in the distorted target class where the distortion is the difference between the data before encoding to the data after decoding. Unless the undesired or the redundant features are removed, an accurate representation of the CoI cannot be realized after the decoding. Therefore, if a zero distortion is expected during the extraction of the target class then, the features which could contribute the most towards the accurate representation of the target class is the need of the hours and this leads to the foremost objective of finding the optimal subset of features to and then followed by an appropriate compression of the data on the given reduced feature space.

guided feature subsetting carried out by preserving the features with low variance as the desired optimal subset of features and the results reported from the work shows the improved compactness within the target class and scatterness among the samples in an unknown population. Utilizing this, the proposed compression scheme is exclusively designed to encode the high dimensional data represented in a feature sub space obtained from the knowledge of the CoI.

It could be observe that the projection of the high dimensional data like hyperspectral remotely sensed data on to a target class guided features subspace that consists of several classes, the compactness of the target class is expected to be maximized when the input pattern is projected on to the target class derived feature subspace due to the elimination of the high variance features. Another expected impact on the data in the intrinsic subspace is the increased variability among the other samples. For such a pre-processed data, compression cannot be based on a common encoding technique instead a hybrid approach is appropriate. The involvement of this research work in the process of compression is at two levels. Employing the vector quantization, a hybrid encoder is designed to generate a primary and secondary code vectors using K-means and ISODATA clustering techniques respectively and a decoder is developed as just a look-up function. The primary code vectors are generated to build the corresponding code book from the target class training set by adopting the K-Means clustering. Similarly, a secondary codebook is developed by adopting the ISODATA clustering

of the input samples that are classified as the counter target class.

The rest of the paper is organized in the following way. The related work is discussed in section 2. The proposed target class focused compression encoding using vector quantization is given in section 3. The experimental results and comparison with existing techniques are presented in section 4. Conclusion is presented in section 5.

II. RELATED WORK

Depending upon the quality of the data resulted after the decompression process the compression techniques can be categorized as lossless, lossy or near-lossy [2]. Obtaining the same decompressed pattern as the original input pattern indicates the effectiveness of the compression algorithm and such lossless compression is well suited for text[3] and audio[4] compression domain. On the other hand, decoding the data from the compressed pattern which is an approximation of the original pattern indicates the information loss [5]. Such a lossy compression algorithms are popularly being used to encode image and video signals [6]. To minimize the information loss but without reproducing the exact version of the original pattern, a few compression algorithms are designed which are known as near-lossy techniques [7].

Basically the selection or the design of a compression technique do not just depend on the time taken for coding and decoding or the achievement of the best compactness of the data but it also depends on the circumstance and the type of the data to be compressed. JPEG is the most popularly used lossy compression technique based on Discrete Cosine Transformation (DCT) that are extensively been used in image compression [8]. A new international standard for lossless data compression is JPEG 2000 which is based on wavelet transformation that provides the higher performance of the compression [9] and is the state-of-the-art also. In order to selectively encode just the target class from among all the other classes present in the input for its lossless decoding, it is necessary to modify the DCT which is not simple. While, the JPEG2000 supports region of interest based compression and is a lossless technique by nature; however it is not suitable for the lossless encoding of the target class because it is unrealistic to expect a region which could cover all the target class samples that are spatially spread. Also, estimating the region which could cover all the samples belonging to the target class itself could be taken as other research work.

Vector Quantization (VQ) is an efficient lossy data compression technique utilized in various applications. Linde, Buzo and Gray (LBG) is a well known K-means variant VQ compression [10]. Hitherto several attempts are mode towards customizing the encoding scheme as required by the circumstances [11]. In the literature, the different types of VQ techniques such as Classified VQ [12], and Adaptive VQ [13] have been reported for various purposes. Vector quantization based image compression [14] exhibits good compression performance at low bit rates while computational complexity has been always a difficult problem for its time consuming encoding system however, the decoding process is very simple. A Transformed Vector Quantization (TVQ) is also reported by combining both transform coding and VQ that takes advantage over VQ [15]. It is also possible to modify the encoding scheme of the VQ such that a lossless compression could also be achieved with respect to the target class by appropriately choosing the prototype vectors and a lossy when concerned to other classes. Therefore, the proposed hybrid compression is designed using a vector quantization on the reduced feature subspace by determining the appropriate prototypes.

Keeping the lossless decoding of the target class as the main objective boundary estimation of the target class is used to define the primary code vectors. On the other hand the secondary code vectors are determined based on the clustering. K-means is one of the most simple and commonly used clustering algorithms but require the K value to be defined and fixed prior to the execution of the algorithm. The Linde, Buzo and Gray (LBG,) vector quantization algorithm which is an alternate to K-means although converges in a finite number of iterations, yet it is NP complete. ISODATA[16] is a generalization of K-means which allows splitting, merging and eliminating clusters dynamically [17] which might lead to better clustering and eliminate the need to set K in advance however, is computationally expensive and is not guaranteed to converge. Although, both the K-means and ISODATA are iterative clustering algorithm but K-means is suitable when the data is highly compact and ISODATA is shown to be effective when the data distribution has high variation. Therefore the hybrid compression uses, K-means clustering to find the optimal code vectors from the target class that are expected to be compact in the reduced space and ISODATA is chosen to build secondary code vectors from the input space the which could have large variation in it.

III. PROPOSED TARGET CLASS GUIDED COMPRESSION

In the process of developing a hybrid compression method, two issues need to be solved. The first issue is to decide the optimal prototypes for the distortion free decoding results of the target class and the second problem is to decide the sub-optimal prototype for the encoding of the other classes. The first issues articulate the primary code book generation while the second issue focuses on the secondary code book generation. The schematic block diagram of the proposed hybrid compression is as shown in figure2. It consists of three components namely, design of the codebook, encoder and the decoder. Given the training set of the CoI as $\mathbf{T_C} = \left\{ \mathbf{s_1}, \mathbf{s_2}, \ldots, \mathbf{s}_t \,\middle|\, s_i \in \Re^N \right\}$ the optimal subset of the features is determined using the method proposed by Nagabushan et al. [1]. As a result, N features get reduced to n which could accurately project the CoI and are used to handle the high dimensional input provided for compression. The input $X = \left\{ x_1, x_2, \ldots, x_{\mathbf{M}} \,\middle|\, x_i \in \Re^N \right\}$ is pre-processed

by projecting it onto the n dimensional feature space derived by the CoI so that the compactness among the CoI samples present in the input space get maximized. Later, the input in the reduced feature space is classified to separate the CoI samples from among other unknown classes nevertheless the process of the classification is not in the scope of this paper. The results of such classification are considered as the input for the design of the code book in the process of hybrid vector quantization. Let $\{T\}$ and $\{\tilde{T}\}$ represent the set of samples classified as target class and counter target class. The main factor that decides the efficiency of the vector quantization is the initialization of the codebook. Suppose K is the total number of the code vectors that collectively outline the codebook then, the larger value of K indicates the less distortion but increases both the searching time required by the encoder and the look up time required by the decoder. Therefore, there is always a trade-off in choosing the number of code vectors. Thus the total code book size $K = k' + k''$ where the primary code book size k' that represent the target class and k'' that represent the other classes are chosen to balance this trade-off.

at the time of classification some of the samples from the unknown classes would have got classified falsely as the target class if in case the input space contains some unknown classes that are characteristically overlapping with the target class although the feature subsetting is so designed to produce the optimal feature set. Hence the initialization of the primary codebook is build from the available training set that is sufficient enough to decide the boundary of the target class. At the same time, as the number of classes present in the input space or the training set of these classes are unknown, the secondary codebook is designed based on the samples classified as the counter target class $\{\tilde{T}\}$. However, the classification errors that might exists in the $\{\tilde{T}\}$ is ignored while designing the secondary codebook.

The study on the impact of the classification error due to either the feature subsetting or the classification process, on the code book design itself could be taken as another detailed future work.

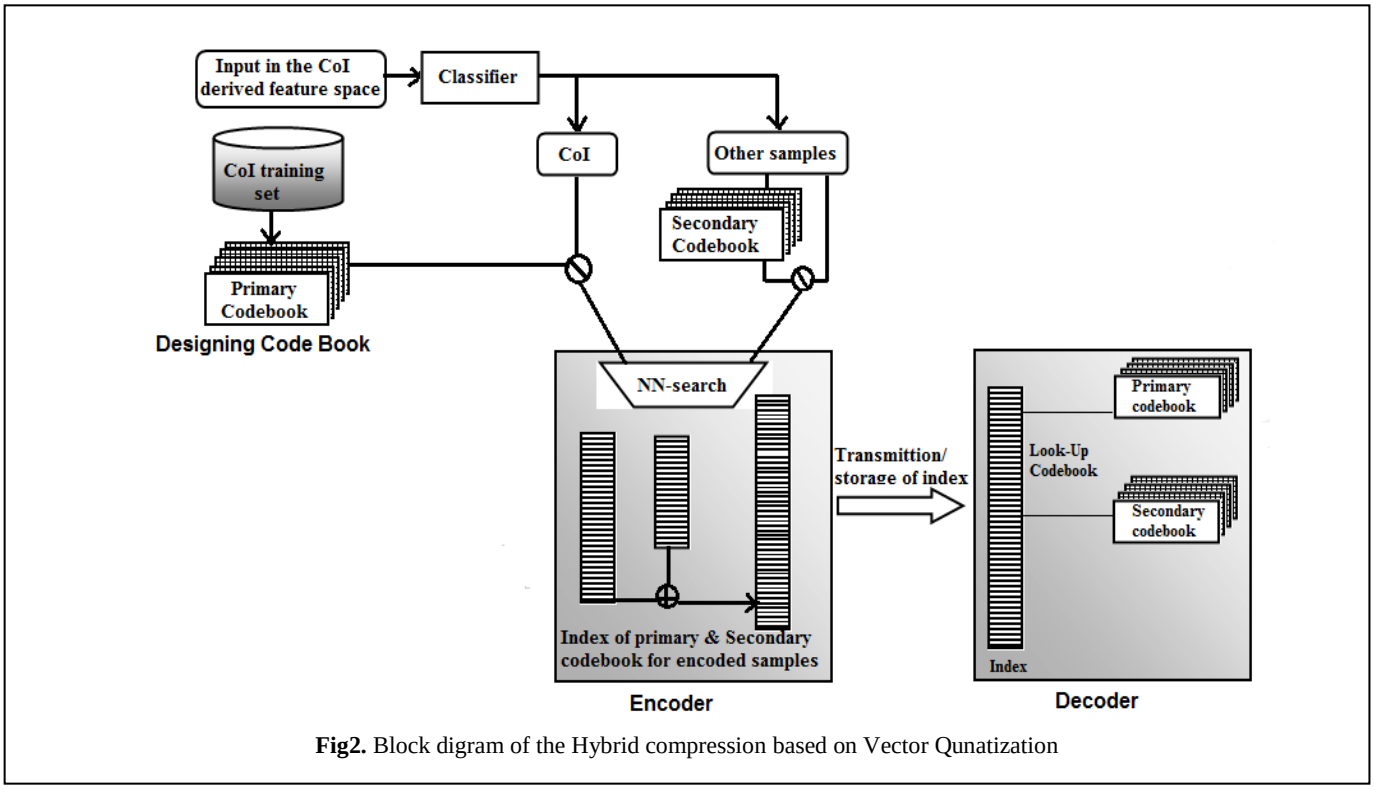

Fig2. Block digram of the Hybrid compression based on Vector Qunatization

A. Design of the code book

Considering the lossless recovery of the target class as the main objective of the hybrid compression scheme, collectively the code vectors should ensure that none of the input samples from the unknown classes ever get encoded by the prototypes chosen from the target class. However it could be possible that,

To design the code book the initial code vectors or code words requires to be estimated. In the literature, various techniques for initializing the code book are available that use the entire training set or assume probability distribution as in case of generalized Lyod algorithm [18]. Linde Buzo Gray uses perturbation while using the splitting technique for initializing

the code book. This technique begins with one-level vector quantizer by considering the training set and continues to build a desired level of quantizer using splitting technique.
The final codebook so obtained is guaranteed to be at least as good as previous stage.
However, there is no guarantee that the procedure will converge to the optimal solution and also does not intend for lossless recovery of data. W.H.Equitz [19] introduced a method for generating an initial codebook called pair wise nearest neighbor but it turns out to be ineffective as it does not focus on the requirement of the CoI. While the hybrid compression is concerned here, the initialization process varies for K-means and ISODATA.

Primary codebook design

To accomplish the lossless decoding of the target class, a variant of the K-means clustering is used. The initial perception that appears is, to consider the mean of the training samples as the only code vector so as to accomplish a highest compression ratio but there could be a tendency of the boundary samples of the target class attempting to magnetize the samples from other unknown classes that could be characteristically similar to it. Consequently, the primary code book is designed starting from the boundary samples of the target class so as to avoid the possibility of the distortion that might occur in it. The boundary samples are those that are more deviated from the mean of the target class. Certainly, the scatterness within the target class might have got minimized when it is represented in the feature subspace that preserves the low variance [1] however, to avoid the possibility of the error that might occur in the encoding of the boundary located samples, primary code vectors are generated by finding the centroids of the clusters formed starting from the boundary samples and iteratively progressing towards the mean.

Let, $\mathbf{T_C} = \left\{ \mathbf{s_1}, \mathbf{s_2}, \ldots, \mathbf{s}_t \,\middle|\, s_i \in \Re^n \,\middle|\, n << N \right\}$ represent the target class training set in intrinsic subspace. The training samples whose Mean Square Error (MSE- Euclidian distance between the mean and the sample) is greater than the pre-estimated threshold are considered as the boundary samples and are used to initialize the clusters centroids. Next, each training sample is compared against each cluster center. It is assigned to a cluster *j* if has the smallest distance with center of *j*. Once, all the samples are assigned to any one of the cluster, then mean of each cluster is re-computed. The detailed procedure is as follows:

TABLE I.
ALGORITHM TO COMPUTE THE PRIMARY CODE VECTORS

Input:	$\mathbf{T_C} = \left\{ \mathbf{s_1}, \mathbf{s_2}, \ldots, \mathbf{s}_t \,\middle|\, s_i \in \Re^n \,\middle|\, n << N \right\}$ //target class training set in intrinsic subspace.
Output:	$\Omega = \{ \vartheta'_1, \vartheta'_2, \ldots, \vartheta'_{K'} \}$ // Primary code vectors
Step1:	**for** i = 1 to t **do**
Step2:	$\{B\} = \max\left(\left\| s_i - \mu_T \right\| \middle| i = 1 \ldots w \right)$ //Compute boundary samples of the target class T_C
Step3:	**end for**
Step4 :	$b = \lvert B \rvert$
Step5:	**for** i=1 to b **do**
	$\mu_b \leftarrow \{B\}$ //initialize the boundary samples as the initial centroid
	end for
Step6:	**Repeat**
Step7	**for** j =1 to t **do**
Step8	$z_j \leftarrow \arg\min_b \left\| \mu_b - s_j \right\|$ // assign target class sample to the nearest boundary
Step9	**end for**
Step10:	**for** i =1: b
Step11:	$C_i \leftarrow \{ s_j : z_j = i \}$
Step12:	$\mu_i \leftarrow mean(C_i)$ // re-compute the mean
	end for
	Until no change in the mean
Step 13	Return $\Omega \leftarrow \mu_i \forall i = 1 \ldots b$

The algorithm converges when no more changes are possible. All the centroids are considered as the primary code vectors and represented as $\Omega = \{ \vartheta'_1, \vartheta'_2, \ldots, \vartheta'_{K'} \}$. The target class samples when mapped to any one of these vector is expected to minimize the distortion. Distortion is the squared Euclidean distance between the samples to be encoded to the code vectors represented as in (1).

$$d^2(s_i, \vartheta_i) = \min d^2(s, \vartheta_j) \forall j = 1 \ldots K \qquad \text{eq(1)}$$

The primary code vectors can be validated by combining the secondary code vectors.

Secondary codebook design

The main objective of generating the secondary code vectors is to avoid the false encoding of the counter target class samples $\{\tilde{T}\}$ with the primary code vectors.

Therefore, appropriate secondary code vectors also play a role in reducing the distortion of the target class which in turn avoids the false acceptance in to the target class when the decoded samples are subjected for classification. Therefore,

the design of the secondary code book is very crucial in deciding the loss free target class. It is not preferable to choose the same procedure as the primary code book design while building an effective secondary code book since; the data distribution details like the total number of classes, overlapping classes in $\{\tilde{T}\}$ are unknown. Also, the $\{\tilde{T}\}$ samples are likely to be very much scattered which makes the boundary estimation technique to be impractical as an alternative, splitting and merging of the clusters could be possible. This brings the necessity of an appropriate clustering technique that might help in the prediction of the appropriate vectors which could encode the other input samples without overlapping with the target class and should also converge with the less number of iteration. The convergence of the ISODATA clustering is slow[20] but its philosophy of splitting and merging of the clusters help in handling the variability in the data. $\{\tilde{T}\}$ are considered as the input for the basic nonparametric ISODATA algorithm to find the centroids of the clusters $K''|K'' << m$ to generate the secondary code vectors and the description is as given in table II. Further, the combined code book symbolized as $\Theta = \{\Omega \cup \Phi\}$ is taken for validation. The input $X \in \Re^n$ is mapped to the codebook Θ to find their nearest code vector and reclassified after the decoding to measure the false acceptance and false rejection rate of the target class. Suppose, the target class is found to have a false acceptance in to it then, the secondary code vector is increased by one. Similarly to avoid the false rejection from the target class primary code vector is increased by increasing the boundary sample of the target class.

TABLE II.
ALGORITHM TO COMPUTE THE SECONDARY CODE VECTORS

Input:	$\sigma^2{}_{\max}$ and k_0 // splitting and merging user specified parameters.
Output:	$\{\tilde{T} \in \Re^n\}_1^m$ // samples classified as counter target class $\Phi = \{\vartheta_1'', \vartheta_2'', \ldots, \vartheta_{K''}''\}$ // Secondary code vectors
Step1:	$k = k_0$ // initialize the mean
Step2:	if $d(\tilde{s}_i, \mu_j) = \min\{(\tilde{s}_i, \mu_1), \ldots (\tilde{s}_i, \mu_{k_0})\}$ then, $k_i \leftarrow \tilde{s}$ //assign the data to cluster
Step3:	if $n_j < n_{\min}$ then discard the j^{th} cluster $k = k-1$ and re-compute the mean
Step4:	if $\sigma^2{}_j >= \sigma^2{}_{\max}$ then split the cluster and re-compute the mean
Step5:	if $k < k_0/2$ then merge the cluster and re-compute the mean
Step6:	$\Phi = \{\mu_1, \ldots \mu_k\} = \{\vartheta_1'', \vartheta_2'', \ldots, \vartheta_{K''}''\}$

Compression by quantization

The compression of the input is defined by an encoder which consists of two mapping functions defined as $f_1(\{T\}) \rightarrow \{\Omega\}$ which maps the samples classified as target class to primary code vectors and $f_2(\{\tilde{T}\}) \rightarrow \{\Phi\}$ that map the counter target class samples with the secondary code vectors such that both the functions should ensure the minimum distortion. The encoding of the input vector x_i using the proposed algorithm doesn't require an exhaustive search over the entire codebook like the conventional vector quantization technique, instead either the primary or secondary codebook need to be searched depending upon the classification results. Each nearest code vector then is assigned with an index-***i*** to form an index table. The indices along with both the code books have to be transmitted or stored required for the decompression. The total number of bits required while representing the code book along with the number of indices required in the proposed algorithm is given in (1).

$$Bitrate = \left(\frac{\log_2 k' + \log_2 k''}{n} \right) \text{bits} \qquad (1)$$

The quantization is dependent on the total number of prototypes being generated and the dimension of each vector. In the original feature space of N-dimension, the time required to search the code book of size K is $O(KN)$. Since, the feature subsetting has reduced the N dimension to n dimension, the decoding rate turns out to be $O(Kn)$.

IV. EXPERIMENTS AND RESULTS

In order to test the feasibility and the performance of the proposed algorithm, some experiments are carried out on the selected two hyperspectral data sets namely AVIRIS Indiana Pines and ROSIS Pavia University[20]. Both the data sets consists of large number of samples with each pixel described in high dimension (>100) and hence there is a scope for experimenting the proposed hybrid compression on the feature reduced space. Since both the data set consists of sufficient number of classes (>5) that are spatially spread, implementation of the target class based compression is possible. From both the data set, only few classes that are spatially spread and are comparatively large in number are chosen as the target class for the experimentation purpose.

TABLE III.
GROUND TRUTH OF THE AVIRIS INDIANA PINE AND PAVIA UNIVERSITY SCENE WITH THEIR RESPECTIVE SAMPLES COUNT

#	AVIRIS Indiana Pine data set		ROSIS Pavia University data set	
	Class	Samples	Class	Samples
1	Alfalfa	46	Asphalt	6631
2	Corn-notill	1428	Meadows	18649
3	Corn-mintill	830	Gravel	2099
4	Corn	237	Trees	3064
5	Grass-pasture	483	Painted metal sheets	1345
6	Grass-trees	730	Bare Soil	5029
7	Grass-pasture-mowed	28	Bitumen	1330
8	Hay-windrowed	478	Self-Blocking Bricks	3682
9	Oats	20	Shadows	947
10	Soybean-notill	972	-	-
11	Soybean-mintill	2455	-	-
12	Soybean-clean	593	-	-
13	Wheat	205	-	-
14	Woods	1265	-	-
15	Buildings-Grass-Trees-	386	-	-
16	Stone-Steel-Towers	93	-	-

Indiana pine data set captured from the AVIRIS sensor has 145x145 pixels, with each pixel represented with 220 spectral bands and for experimentation calibrate 200 bands are used. Pavia University data set is a scene captured over Pavia by ROSIS sensor which has 610x610 pixels and has 103 spectral bands. The classes with small count in the samples were not tested as target class but were considered during the encoding and decoding process. The classes that were selected as the target class in the separate experimentation are as highlighted in the table III.

A. *Pre-processing of the data*

Experimentation was carried out separately from each data set considering just one class as the target class at a time. From Indiana pine data set corn-notill, corn-mintill, soyabean-mintill, Grass trees and woods; and Asphalt, Meadows, Bare soil class were considered as the target class. The experiment was carried out on these selected 9 classes as target class. For each target class feature subsetting was employed based on the procedure described by [1].

Corn_notill as target class: Considering corn-notill as the target class the experimental procedure is as explained. 30% out of 1428 samples were randomly selected as the training samples and 78 spectral bands with large variance which were tending to split the corn-notill class were eliminated.

These remaining spectral bands were used to find the optimal subset of features which can project the corm-notill class. Using the compactness among the target class samples as a measure 37 features were validated and [103-108],[142-148], [151-153]and [180-200] were accepted as the optimal subset. Further, all the 21025 high dimensional input samples were characterized in the target class decided 37 spectral bands and the remaining bands were discarded. The input samples represented in the intrinsic subspace as defined by the corn_notill class when classified to extract the corn-notill then 1020 samples were falsely accepted as corn-notill, 308 samples were falsely rejected from the target class. These results were further used to design the code book.

B. *Compression guided by the target class*

1. *Primary code book generation:* 30% of the corn_notill was randomly selected as the training set to get the primary code vectors employing the K-means clustering algorithm. Since, the corn_notill target class was normally distributed, initially 4 boundary samples were considered as the prototypes.

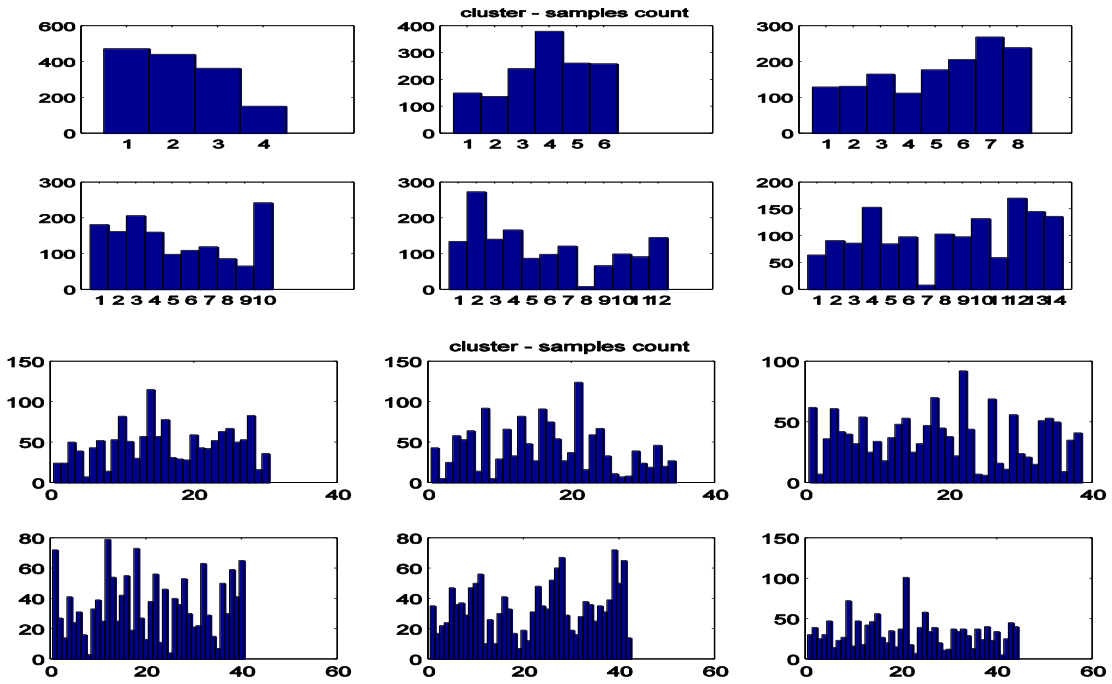

Fig3. Histogram plot of the initialization procedure of the primary code book based on different boundary samples chosen

The boundary samples from the training samples of the Corn_notill clas was computed as in (2) where μ is the initial mean of the t training samples.

$$\{B\} = \max\left(d\left(s_1, \mu\right)\ldots d\left(s_i, \mu\right)\forall i = 1\ldots t\right)$$

eq(2)

Starting from the K=4 boundary samples, the experiment was repeated to up to K=44 boundary samples. Figure 3 shows the histogram plot for the number of corn-notill samples clustered to their nearest centroid when boundary samples were varied. Too many samples (>300) were clustered to each prototype when K=4-6 which was found to be gradually balanced by increasing the K value. It was observed that when K=40-60 then each cluster had (<100) target class samples. The optimal spectral bands selection and its utilization in the representation of the Corn_notill class didn't allow more computation required to find the optimal code vectors. The algorithm was converged within 8 iterations. For K=44, the algorithm was converged at 8 iteration and similar results are shown in the figure 4a.

As reported about the classification results earlier, 308 Corn_notill samples were falsely rejected and were classified as other class samples which totally resulted in $\{\tilde{T}\}$ =18,885 samples as counter target class. Since, more samples were available 40% of these samples were randomly chosen to build the secondary code book using the ISODATA algorithm. Number clusters was initialized to be = 50 based on the total number of counter target class to be compressed. Also the user can also initialize the number of clusters based on the data distribution of the samples under $\{\tilde{T}\}$. It was observed that more than 5000 samples were found to be more deviated from the mean which indicates the scatterness of the data and in the normalized scale within $\{\tilde{T}\}$ the σ . >0.891. Similarly to merge the clusters, the merge parameter was based on two strategies. First, the distance between the centroids of the individual cluster is < 0.1 and the second strategy is to check for the number of samples within each clusters >=300. In this experiment, both the strategies were found to be applicable. For K=50 the ISODATA algorithm was converged with 12 iterations as shown in fig 4b.

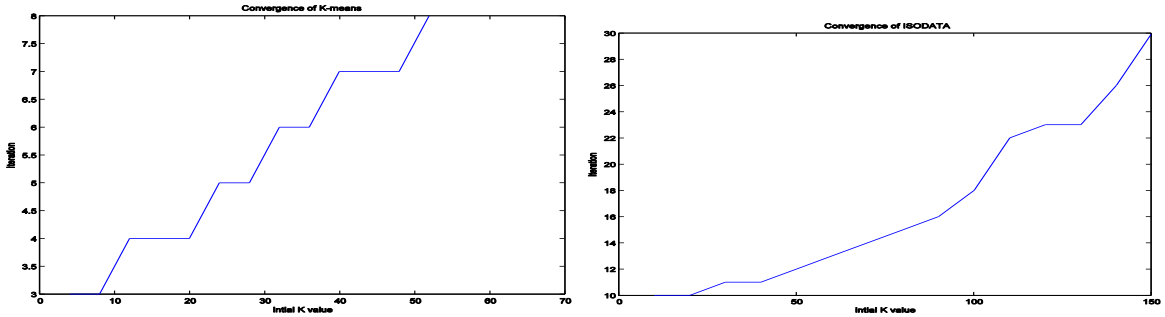

Fig 4a-4b. Convergence of K-means and ISODATA clustering while generating the Primary and Secondary code vectors for the target

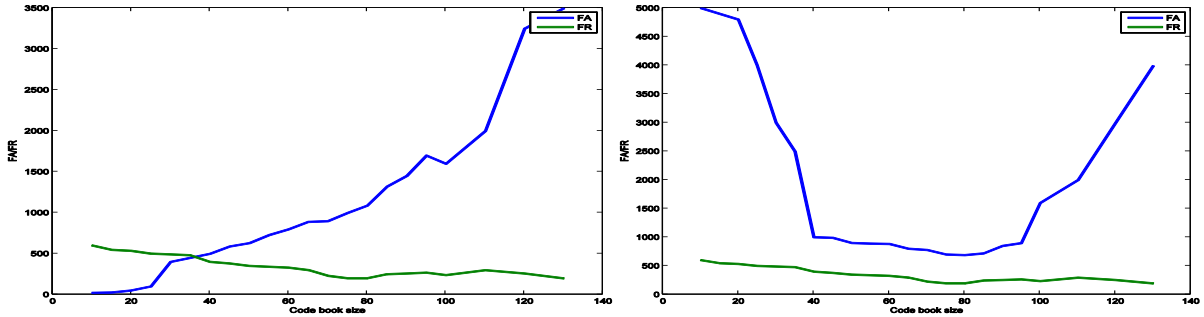

Fig 5a-5b. False acceptance and false rejection rate of target class: corn_notill by varying the code book size

3. Validation of Code vectors:

k' =40 and k'' =50 primary and secondary code vectors were considered together for validation. 21025 input samples that is represented using 37 spectral bands as decided by the corn_notill (including the 1428 corn_notill target class samples) was mapped to K =90 code vectors. For each code vector in K , a sequence number was assigned to generate the index table. Total 1120 samples classified as corn-notill were encoded by mapping with 40 code vectors and 18,825 samples were mapped to 50 secondary code vectors. The mapping was carried out by finding the nearest code vectors using the Euclidian distance. Later, the encoded data was decoded using the combined code book of size 90 vectors and then it was reclassified to measure the classification results of corn_notill. It was observed that 1400 samples were falsely classified as corn_notill and 390 samples were false rejected from corn_notill class. This gives rise to the further correction of the code book. Thus the experimentation was repeated by increasing k' and k'' . Fig5a & 5b plots the classification results by varying the code book size. It can be observed that even after increasing both the code book sizes neither the false rejection nor the false acceptance was unavoidable completely. The reason may be due to the overlapping classes that could be characteristically similar to corn_notill class which needs to be explored.

Indiana pine data set has 145x145 pixels and each pixel has 200 spectral bands where each pixel takes 14 bits. Therefore, the storage space required if it is stored in the original form is 145x145x200x14bit= 57490KB. If the same data is compressed then, 200 spectral bands are reduced to 37 bands 145x145 samples are quantized by 90 code vectors. i.e., 90x 37x14 bit= 45 KB.

Further, increasing the primary and the secondary code vectors to optimize the classification results of the target class, the bit rate of the compression found to be decreased but was less than the original data.

Similar experimentation was carried out on the selected target class from Indiana pine and Pavia University data set. Several spectral correlations were found to exist in the Pavia University data set, particularly in the Asphalt target class. Fig 6 shows the histogram plot when the primary code book was build using the K-means. It was observed that the 3 classes when chosen as target class and experimented separately, with K=300-1500 primary code vectors the compression of the target was accomplished due to high spatial correlation.

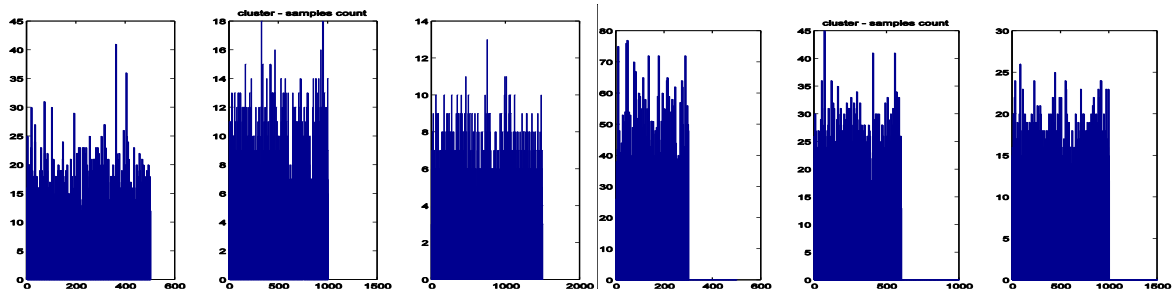

Fig6. Histogram indicating the primary code book initialization varying K=300-1500 for target class Asphalt

Table IV lists the results of the proposed compression when incorporated in the feature subspace. The classification results of the target class are based on the number of samples that are truly classified as target class and number of samples that were falsely rejected for which precision and recall were used. Specificity is also used as a measure to check how many samples are relevant among the samples that were classified as target class. The equation (3) gives Specificity- ρ , Precision- δ and Recall- υ .

$$\text{Re}\,call = \upsilon = TP \big/ (TP + FN)$$

$$Specificity = \rho = TN \big/ (TN + FP) \qquad \text{eq(3)}$$

$$\text{Pr}\,ecision = \delta = TP \big/ (TP + FP)$$

C. ***Comparison of the proposed compression with contemporary compression methods***

The efficiency of the proposed target class guided hybrid compression method was tested with Classified Vector Quantization (CVQ), JPEG based on DCT and JPEG2000 using wavelet transformation. To carry out CVQ the knowledge of the all the class were considered and then subjected for classification. Next, for each class optimal prototypes were selected based on clustering within the class and were considered as encoder.

Next comparison was using DCT is a lossy compression technique. Considering all the 21025 samples of Indiana pine data set optimal features were obtained based on the features that preserve the higher variance. On the reduced space, for a block of 8X 8, JPEG algorithm was implemented.

We observe that the classification accuracy of none of the target class was not as good as the proposed method and also the encoding time was higher.

In continuation to this the state of art, JPEG2000 was also tested on the reduced feature space. Since the target class were scattered at various place locating them was a difficult task. Thus, the entire samples were considered together for compression. The comparative results of the target class are as shown in table V.

TABLE-IV
CLASSIFICATION RESULTS AFTER DECODING

Data set	CoI	Feature Subspace	Compression bit rate	Size of Θ	Encoding Time(sec)	Decoding Time(sec)	ρ	δ	υ
AVIRIS Indian Pines	Corn-notill	37	0.34	90	50	37	0.87	0.78	0.80
	Corn-mintill	46	0.45	120	40	32	0.78	0.89	0.81
	Grass-Pastures	81	0.67	80	50	23	0.91	0.81	0.81
	Soybean-mintill	67	0.71	180	42	14	0.90	0.91	0.78
	Woods	72	0.51	200	60	19	0.81	0.9	0.71
ROSIS Pavia University	Asphalt	25	0.89	150	1800	60	0.91	0.90	0.80
	Meadows	31	0.91	180	3000	60	0.91	0.91	0.82
	Bare soil	28	0.88	200	2500	60	0.91	0.98	0.83

TABLE-V
COMPARISON OF PROPOSED COMPRESSION WITH CONTEMPORARY METHODS

Data Set	Target class	Algorithm	Accuracy	Sensitivity	Precision	Specificity
AVIRIS Indiana Pine data set	Corn-notill	JPEG	85%	0.749	0.898	0.938
		JPEG2000	89%	0.81	0.78	0.79
		CVQ	75%	0.703	0.800	0.891
		Hybrid	80%	0.746	0.833	0.893
	Grass-Pastures	JPEG	63.1%	0.661	0.708	0.804
		JPEG2000	95%	0.899	0.812	0.81
		CVQ	93%	0.900	0.962	0.977
		Hybrid	94.7%	0.903	0.967	0.971
ROSIS Pavia University	Asphalt	JPEG	61.23%	0.498	0.657	0.829
		JPEG2000	82%	0.78	0.71	0.72
		CVQ	72%	0.666	0.932	0.976
		Hybrid	84.2%	0.79	0.923	0.967
	Meadows	JPEG	58.1%	0.445	0.54	0.789
		JPEG2000	81%	0.61	0.81	0.89
		CVQ	78%	0.565	0.862	0.948
		Hybrid	89%	0.653	0.853	0.928

V. CONCLUSIONS

In this paper a new hybrid compression technique is developed and implemented for the encoding and decoding of the target class in the reduced feature space. Due to the optimal feature subsetting carried out based on the knowledge of the target class was used to project the given high dimensional input space. The hybrid compression scheme was developed using the vector quantization which uses two different clustering schemes namely K-means and ISOADAT algorithms.

The compactness realized within the target class were encoded using the primary code vectors generated from the K-means clustering and other samples were encoded using the secondary code book which was developed using ISODATA clustering. In addition to the accomplishment of the compression bit rate the classification results were found to be satisfactory. However, there is a scope for future work in continuation to thus as listed below:

1) It is required to refine the primary if in case the input contains overlapping classes which are characteristically similar to target class.

2) If an application provides the training set of the target class as well as the training set of the class associated with it then, a new method requires to be devised to accurately encode both the target class and its associated class.

3) Since the encoder is dependent on the code vectors being generated, the distortion free target class is dependent on the feature subspace process and classification process. Therefore, the study on the impact of the classification

error due to either the feature subsetting or the classification process on the code book design itself could be taken as a detailed future work.

REFERENCES

[1] P.Nagabhushan, H.N.Meenakshi., Target Class Supervised Feature Subsetting, International Journal of Computer Applications (0975 – 8887) vol.91 , no.12, 2014.

[2] Lei Wang, LichengJiao ,JiajiWu, GuangmingShi,YanjunGong., "Lossy-to-lossless image compression based on multiplier-less reversible integer time domain lapped transform", Signal Processing: Image Communication, vol. 25,pp.622–632,2010.

[3] Dohyoung Lee, Konstantinos N.Plataniotis.,"Lossless compression of HDR color filter array image for the digital camera pipeline ", Signal Processing: Image Communication, vol. 27, pp.637–649, 2012.

[4] W.H.Equitz., *"A New Vector Quantization Clustering Algorithm",* IEEE Transaction on Accoust, speech and signal processing, 1989.

[5] J. Janosky and R.W. Witthus, "Using JPEG2000 for enhanced preservation and web access of digital archives", IST Archiv. Conf., pp.145-149,2004.

[6] Ismael Baeza , José-Antonio Verdoy , Rafael-Jacinto Villanueva , Javier Villanueva-Oller., "SVD lossy adaptive encoding of 3D digital images for ROI progressive transmission", Image and Vision Computing,vol.28,pp. 449–457,2010.

[7] M. Mohamed Sathik, K.Senthamarai Kannan and Y.Jacob Vetha Raj, "Hybrid Compression of Color Images with Larger Trivial Background by Histogram Segmentation", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 8, No. 9, December 2010.

[8] G.K.Wallace, "The JPEG sill picture compression standard", Communications of the ACM, vol. 34, pp. 30-44,1991.

M. Rabbani and R. Joshi, An overview of the JPEG2000 still image compression standard, Signal Processing: Image Com., 17:3-48, 2002

[9] Bryant Aaron, Dan E.Tamir, Naphtali D. Rishe and Abraham Kandel., " Dynamic Incremental K-means Clustering", Proc IEEE International Conference on Computational Science and Computational Intelligence, 308-313, 2014.

[10] N.M. Nasrabadi, R.A. King, "Image coding using vector quantization: a review", IEEE Trans. on Communications, vol. 36, 957-571, 1988.
M. Goldberg, "Image Compression using Adaptive Vector Quantization," *IEEE Transactions on Communications*, vol. 34, no. 2, pp. 180-187, 1986.

[11] Yoseph Linde, Andres Buzo, Robert **m.** Gray., "An Algorithm for Vector Quantizer Design", **IEEE** transactions on Communications, vol.28, no. **1,** pp.84-95,1980.

[12] Krishnamurthy R. and Punidha R.," FVQEOPT: Fast Vector Quantization Encoding with Orthogonal Polynomials Transform", International Journal of Computer Theory and Engineering, vol. 5, no. 1, pp.31-35.2013.

[13] L. Torres and J. Huguet, "An Improvement on Codebook Search for Vector Quantization," *IEEE Transactions on Communications*, vol. 42, no. 234, pp. 208-210, 1994.

[14] M.J. Ryan and J.F. Arnold, "The lossless compression of AVIRIS images by vector quantization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, no. 3, pp. 546–550, 1997.

[15] X. Pan. R. Liu, and X. Luv, "Low-Complexity Compression Method for Hyperspectral Images Based on Distributed Source Coding," IEEE Geoscience and Remote Sensing Letters, 9(2), pp.224 227,2012.

[16] Geoffrey h. Ball and David j. Hall. "ISODATA, A novel method of data analysis and Pattern Classification". Technical Report, April 1965.

[17] Lloyd, ***S.*** P., "Least Squares Quantization in PCM's," Bell Telephone Laboratories Paper, Murray Hill, NJ, **1957**

[18] W. Equitz, "A New Vector Quantization Clustering Algorithm," *IEEE Transactions on Acoustic, Speech, Signal processing*, vol. 37, no. 10, pp. 1568-1575, 1989.

[19] *AVIRIS Images*, Jet Propulsion Laboratory, NASA, http://aviris.jpl.nasa.gov/html/aviris.overview.html. Anil K. Jain, M. N. Murty, P. J. Flynn, "Data Clustering: A Review," ACM Computing Surveys, vol. 31,no. (3), pp.264-323, 1999.