Global Journal of Computer Science and Technology
Graphics & Vision
Volume 13 Issue 7 Version 1.0 Year 2013
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

A New Classification Performance Aware Multisensor, Multi Resolution Satellite Image Compression Technique

By Ch. Ramesh, Dr. N.B. Venkateswarlu & Dr. J.V.R. Murthy
JNTUK University, India

Abstract - Effective utilization of bandwidth and storage space is important in imaging applications including remote sensing. Remote sensing applications use multi-sensory, multi-band, multi resolution images. Usually, remote sensing applications uses image classification results for their analysis and decision making. In this paper we propose a new JPEG based image compression algorithm based on zooming-shrinking technique. Proposed algorithm performance is evaluated in relation to standard JPEG algorithm. In order to envisage the effect of compression on classification performance, Maximum Likelihood, Mahalanobis and Minimum distance classifiers performance is evaluated with original image data, standard JPEG compressed data and the compressed image data with the proposed method. Experiments are carried out with multi-band images with various resolutions. Our experiments supports that the classification accuracies of compressed images are at par with original image data.

Keywords : peak signal to noise ratio (PSNR), zooming, shrinking, maximum likelihood, mahalanobis, confusion matrix, kappa coefficient, cross validation.

GJCST-F Classification: I.4.2

A NEW CLASSIFICATION PERFORMANCE AWARE MULTISENSOR, MULTI RESOLUTION SATELLITE IMAGE COMPRESSION TECHNIQUE

Strictly as per the compliance and regulations of:

A New Classification Performance Aware Multisensor, Multi Resolution Satellite Image Compression Technique

Ch. Ramesh [α], Dr. N.B. Venkateswarlu [σ] & Dr. J.V.R. Murthy [ρ]

Abstract - Effective utilization of bandwidth and storage space is important in imaging applications including remote sensing. Remote sensing applications use multi-sensory, multi-band, multi resolution images. Usually, remote sensing applications uses image classification results for their analysis and decision making. In this paper we propose a new JPEG based image compression algorithm based on zooming-shrinking technique. Proposed algorithm performance is evaluated in relation to standard JPEG algorithm. In order to envisage the effect of compression on classification performance, Maximum Likelihood, Mahalanobis and Minimum distance classifiers performance is evaluated with original image data, standard JPEG compressed data and the compressed image data with the proposed method. Experiments are carried out with multi-band images with various resolutions. Our experiments supports that the classification accuracies of compressed images are at par with original image data.

Keywords : peak signal to noise ratio (PSNR), zooming, shrinking, maximum likelihood, mahalanobis, confusion matrix, kappa coefficient, cross validation.

I. Introduction

In the recent years use of remote sensing satellite data, air borne sensor data for urban monitoring, traffic monitoring, automatic navigation in driverless cars, disaster warning systems, monitoring the movements of terrorists has increased by many fold in addition to conventional applications such as natural resources management .These applications involves acquisition, communication, storage and processing of horrendous number of images of earth surface.

This situation is becoming more aggravated because of increased pixel resolution, gray level resolution, band resolutions and reduced repetition cycle of satellite. All of these development demands more band width for downlink lines of satellite in addition to more disk space for storage.

In communications, data compression techniques under the name hood of image coding are widely used to reduce the communication bandwidth bottlenecks during data communication. For instance, JPEG standard is used for still image compression [1], MPEG is used for video compression [2]. Also, while communicating data from satellites to ground stations some compression methods are used [3].

A typical image processing system is as shown in Figure 1 that is commonly employed for remote sensing applications. It is very common that most of the application scientists using original image data for their processing. In majority of remote sensing applications, results of classification are the ultimate interest [4].

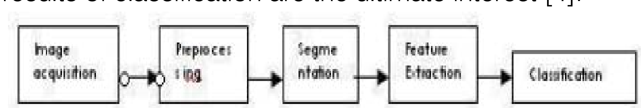

Figure 1 : A Typical Image Processing System

In this study, we propose to study how the classification results will vary if we use compressed image data instead of original image data. Usually applications such as land use classifications assumes samples of a group will be having small random variations in their pixel values while samples of different groups to be having contrastingly different pixel values. Because of the increased pixel and gray level resolutions, samples of a group may be having similar pixel values. Moreover, they will be having high level of spatial auto correlation. Evidently, majority of compression methods exploits this auto correlation to achieve high compression ratios with acceptable PSNR (Peak Signal to Noise Ratio) values [5]. For instance MPEG coding that employs DCT based video compression is widely used for real time surveillance [2].

Our proposed Algorithm is based on Zooming-Shrinking concept. In this Algorithm instead of sending the original image, we send the shrinked image and the difference image between the original image and zoomed image. The size of the shrinked image is very less as compared to the original image that's why it requires less number of bits for communication. The size of the difference image is equals to the original image but the magnitude of the pixel values in the difference image is very low as compared to the magnitude of the pixel values in the original image, that's why it also requires very less number of bits for communication. Therefore the total no of bits required for communicating the shrinked image and the difference image is less than original image. Thus we are achieving compression benefit. We have compared the compression performance of our algorithm with

Authors α σ : Professor, Dept. of CSE, AITAM, Tekkali, A.P, India. E-mails : chappa_ramesh01@yahoo.co.in, venkat_ritch@yahoo.com
Author ρ : Professor, Dept. of CSE, College of Engineering, JNTUK, A.P, India. E-mail : mjonnalagedda@gmail.com

standard JPEG algorithm with variety of images, especially from remote sensing applications.

Also in this study, we evaluate the classification performance of popular classification algorithms like Maximum Likelihood, Mahalanobis and Minimum distance by taking original image data, JPEG compressed image data, compressed data that is compressed by our zooming-shrinking based JPEG image compression method.

Our paper work is organized as follows. Section II introduces the standard JPEG algorithm. Section III explains proposed compression algorithm. Selected classification algorithms are illustrated in section IV. Section V includes details about our experimentations and results. Section VI contains conclusions about our research work.

II. Brief Overview of JPEG Encoding / Decoding System

JPEG is a well known standardized image compression technique. JPEG loses information so the decompressed picture is not the same as the original one. The main reason for use of JPEG is to reduce the size of image files. Reducing image files is an important procedure for transmitting files across networks or archiving libraries. Usually JPEG can remove the less important data before the compression; hence JPEG is able to compress images meaningfully, which produces a huge difference in the transmission time and the disk space. Figure 2 shows the basic Architecture of JPEG compression system. Here is a brief overview of the JPEG compression system. [5]

The image is first subdivided into pixel blocks of size 8X8, which are processed left to right, top to bottom. As each 8X8 block or sub image is encountered, its 64 pixels are level shifted by subtracting the quantity L/2, where L is the Gray level resolution of the image . The 2-D Forward Discrete Cosine Transform (FDCT) (Eq-1)[5]of the block is then computed, quantized using 64 corresponding step size values from the quantization table in Figure 3[6]. After quantization the DCT coefficients are rearranged in a zigzag sequence order as shown in the Figure 4. [6]

Since the one-dimensional reordered array generated under the zigzag pattern of Figure 3 is qualitatively arranged according to increasing spatial frequency, the JPEG coding procedure is designed to take the advantage of the long runs of zeros that normally result from the reordering. In particular, the nonzero AC coefficients (the term AC denotes all transform coefficients with the exception of the zeroth or DC coefficient) are coded using a variable-length code that defines the coefficient's value and number of preceding zeros. The DC coefficient is difference coded relative to the DC coefficient of the previous sub image.

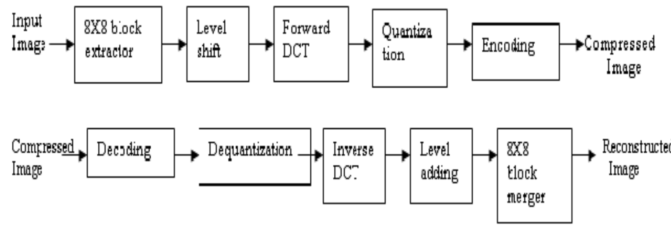

Figure 2 : Basic Architecture of JPEG Compresssion

The 2-D DCT is

$$C(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (1)$$

for u, v = 0, 1, 2, ... , N -1

$$\alpha(u) = \begin{cases} \sqrt{1/N} & for \quad u = 0 \\ \sqrt{2/N} & for \quad u > 0 \end{cases} \quad (2)$$

$$\alpha(v) = \begin{cases} \sqrt{1/N} & for \quad v = 0 \\ \sqrt{2/N} & for \quad v > 0 \end{cases} \quad (3)$$

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 3 : Quantization Matrix [6]

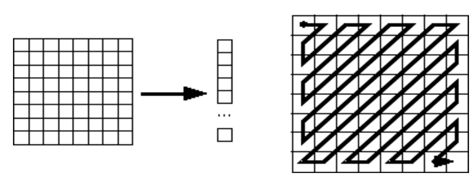

Figure 4 : Zigzag Séquencé [6]

The decompression process performs an inverse procedure. It decodes the Huffman codes. Then, it makes the inversion of the Quantization step. In this stage, the decoder raises the small numbers by multiplying them by the quantization coefficients. The results are not accurate, but they are close to the original numbers of the DCT coefficients. An Inverse Discrete Cosine Transform (IDCT) (Eq.4) [6] is performed on the data received from the previous step. Finally add L/2 to each sub image. Place the sub images in their correct positions.

$$\hat{f}(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \alpha(u)\alpha(v)C(u,v)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (4)$$

The error between the original image and reconstructed image is calculated in terms of Peak signal to noise ratio

$$PSNR = 10 \log_{10} (L^2/MSE) \qquad (5)$$

$$MSE = \frac{\sum_{x=0}^{m-1}\sum_{y=0}^{n-1}\left[\hat{f}(x,y) - f(x,y)\right]^2}{mXn} \qquad (6)$$

MSE – Mean Squared Error

$\hat{f}$(x, y) - Reconstructed Image

f (x, y) – Original Image

m x n – Size of the Image

III. Zooming-Shrinking based JPEG Algorithm

Zooming is a method of increasing the size of a given image. Zooming can be viewed as oversampling or up sampling of a given image. Zooming requires two steps: the creation of new pixel locations, and the assignment of gray levels to those new locations. Shrinking is a method of decreasing the size of a given image. Shrinking can be viewed as under sampling or down sampling. Interpolation is a basic tool used extensively in zooming and shrinking tasks interpolation is the process of using known data to estimate values at unknown locations. Widely used interpolation algorithms are nearest neighborhood, bilinear and bicubic. The bicubic method is more accurate than nearest neighborhood and bilinear interpolation [6]. The proposed algorithm uses the bicubic interpolation. The Proposed Algorithm is described in four stages. The same is illustrated in Figures 5(a)&(b), 6(a)&(b), 7(a)&(b), 8(a)&(b).

Algorithm:

Stage 1

1. Shrink the input image M (Shrinking factor) times, the resulting image is called shrinked image.
2. Zoom the shrinked image M (Zooming factor) times, the resulting image is called Zoomed image.
3. Find the difference image between the Zoomed image and the input image, the resulting image is called the Difference image.

Figure 5(a)

Figure 5(b)

Stage 2

1. The shrinked image is first subdivided into pixel blocks of size 8x8 which are processed from left to right, top to bottom.
2. For each block its 64 pixels are level shifted by subtracting the quantity L/2 where L is the gray level resolution.
3. The 2D-DCT of each block is computed.
4. Quantize the DCT blocks by standard quantization matrix.
5. Form a 1-D sequence of Quantized Coefficients by using Zigzag pattern.
6. Coding the coefficients using JPEG Huffman tables.
7. The receiver decodes the received codes and forms the reconstructed shrinked image.
8. Zoom the reconstructed shrinked image M times

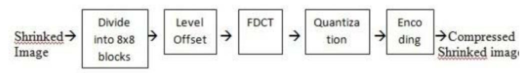

Figure 6(a) : Encoding

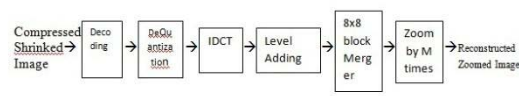

Figure 6(b) : Decoding

Stage 3

1. The difference image is first subdivided into pixel blocks of size 8x8 which are processed from left to right, top to bottom.
2. For each block its 64 pixels are level shifted by subtracting the quantity L/2 where L is the gray level resolution.
3. The 2D-DCT of each block is computed.
4. Quantize the DCT blocks by standard quantization matrix.
5. Form a 1-D sequence of Quantized Coefficients by using Zigzag pattern.
6. Coding the coefficients using JPEG Huffman tables.
7. The receiver decodes the received codes and forms the reconstructed difference Image.

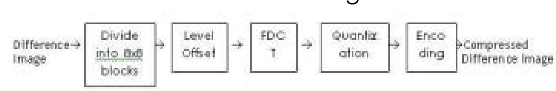

Figure 7(a) : Encoding

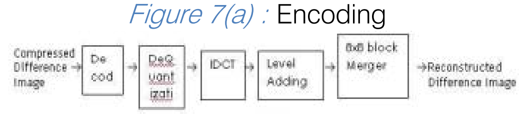

Figure 7(b) : Decoding

Stage 4

1. Add the reconstructed zoomed image obtained at stage 2 and the reconstructed difference image obtained at stage 3 and the resulting image is called output image.
2. Compute the error (in terms of PSNR) between the input image and output image.

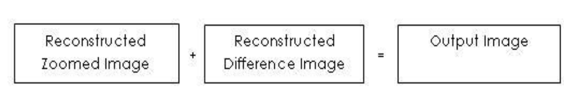

Figure 8(a)

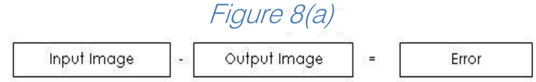

Figure 8(b)

In this algorithm instead of sending the original image, we send the shrinked image and the difference image between the original image and zoomed image. The size of the shrinked image is very less as compared to the original image that's why it requires less number of bits for communication. The size of the difference image is equals to the original image but the magnitude of the pixel values in the difference image is very low as compared to the magnitude of the pixel values in the original image, that's why it also requires very less number of bits for communication. Therefore the total no of bits required for communicating the shrinked image and the difference image is less than original image. Thus we are achieving perceivable compression benefit.

IV. Popular Classification Algorithms

a) Maximum Likelihood Classifier

Let w_1, w_2, . ., w_m denote m distinct populations (classes) with known d-dimensional probability density functions $p_1(X)$, $p_2(X)$, ... $p_m(X)$, respectively. The a priori probabilities that an observation is selected from populations w_1, w_2, . . . , wm are denoted by q_l, q_2, ..., q_m, respectively[7]. According to the Bayesian ML classification rule, assuming equal costs for misclassifications, a random d dimensional pixel vector X is classified as class wk if

$$q_k p_k(X) = max\{q_i P_i(X)\} \text{ for } i = 1, 2 \ldots\ldots, m. \tag{7}$$

Assuming equal a priori probabilities for all the classes, decision rule (7) becomes:

$$X \in w_k \text{ if}$$

$$p_k(X) = \max\{p_i(X)\}, i = 1, 2, \ldots, m. \tag{8}$$

In Equations (7) and (8), the probability density $p_k(X)$ will be given as:

$$p_k(X) = \frac{1}{(2\pi)^{d/2} \left| \sum_k{}^{1/2} \right|}$$

$$X \exp[-1/2.(X - M_k)^T \sum\nolimits_k^{-1} (X - M_k)]. \tag{9}$$

Here, M_k and Σ_k, are the mean vector and covariance matrices of the k^{th} class, and are calculated from the training data. Σ_k is a symmetric positive definite matrix. $\sum_k^{-1}, \left|\sum_k\right|$ are the inverse and determinant of the covariance matrix Σ_k.

b) Mahalanobis Classifier

According to this classifier a d-dimensional random pixel vector (X) will be assigned to the group to which it is nearest [8]. Each group is characterized by its mean vector, which is calculated from training data. Nearness is determined by the Mahalanobis distance between the group mean and X. In mathematical terms, the same classification rule can be represented as:

$$X \in w_i \tag{10}$$

Where i = (1, 2, . . . C) groups if $d_i(X) < d_j(X)$ for all $j \neq i$ where

$$d_i(X) = (X-M_i)^T \Sigma^{-1} (X-M_i) \tag{11}$$

and Mi is mean vector of i^{th} group indicates vector should be transposed. Σ^{-1} is the inverse of the pooled covariance matrix Σ.

c) Minimum Distance Classifier

According to this classifier a random ddimensional pixel vector (X) will be assigned to the group to which it is nearest [9]. Each group is characterized by its mean vector, which is calculated from training data. Nearness is determined by the Euclidean distance between the group mean and X. In mathematical terms, the same classification rule can be represented as:

$$X \in w_i \tag{12}$$

Where i = (1, 2, . . . C) groups if $d_i(X) < d_j(X)$ for all $j \neq i$ where

$$d_i(X) = (X-M_i)^T(X-M_i) \tag{13}$$

and M_i is mean vector of i^{th} group. T indicates vector should be transposed.

V. Experimentations and Results

For the purpose of experimental work, Landsat TM data from USGS data base "www.usgs.gov" is used. Experiments are carried out under MS Windows XP version 2002, SP3 edition. The experimental system is equipped with Intel core 2 Duo 2.60 GHz processor with 1 GB RAM. Using ERDAS Imagine 8.6 (copy rights©1991-2002, Lieca Geo systems) Training sites are labeled. Programs are written in C language under Microsoft Visual Studio 2005 version 8.0.

We have carried out extensive simulations with the selected images and proposed algorithm with different M values. M=2 results the better PSNR. Table 1 shows the Compression Benefit and PSNR loss of proposed algorithm with M=2 as compared to the standard JPEG compression. With all the images we found that proposed algorithm have better compression ratios as compared to standard JPEG coding. The PSNR loss in proposed algorithm is very small as compared to standard JPEG compression. Figure 9 shows the sample (Mono_lake image) Original image, JPEG compressed image, Proposed compressed image.

Original Image JPEG compressed image Proposed compressed image

Figure 9

Table 1 : Compression benefit and PSNR loss of proposed algorithm compared to standard JPEG

Image	Size	No. of bits required for standard approach	No. of bits required for our approach	% of Saving	PSNR standard approach	PSNR for our approach	% loss in PSNR
Mono_lake-band1		410486	260010	36.65	39.871	36.463	8.547
Mono_lake-band2		306006	215852	29.46	42.237	39.111	7.401
Mono_lake-band3		399635	250777	37.24	40.435	37.001	8.492
Mono_lake-band4	1040 X 1040	366063	244232	33.28	41.097	37.443	8.891
Mono_lake-band5		578965	339670	41.33	37.724	33.517	11.152
Mono_lake-band6		249582	221937	11.07	46.608	39.658	14.911
Mono_lake-band7		439079	270574	38.37	39.828	36.139	9.262
		2749816	**1803052**	**34.430**	**41.114**	**37.047**	**9.892**
P143r49_5t19921106_nn1		22276827	13753318	38.261	38.468	35.335	8.144
P143r49_5t19921106_nn2		17114083	11508975	32.751	41.309	38.536	6.712
P143r49_5t19921106_nn3	8139 X 7186	26101868	15407693	40.970	37.970	34.533	9.051
P143r49_5t19921106_nn4		31070502	18600285	40.135	36.534	32.674	10.565
P143r49_5t19921106_nn5		38128640	23225187	39.087	34.765	30.755	11.534
P143r49_5t19921106_nn6		13264142	10952351	17.428	44.396	40.921	7.827
P143r49_5t19921106_nn7		33122366	19806016	40.203	36.005	33.694	6.418
		181078428	**113253825**	**37.445**	**38.492**	**35.206**	**8.536**
P143r51_5t19910410_nn1		21219663	12918117	39.12	39.182	35.892	8.396
P143r51_5t19910410_nn2		17565849	11399598	35.103	41.166	38.561	6.328
P143r51_5t19910410_nn3		29131208	15999585	45.077	37.159	33.756	9.157
P143r51_5t19910410_nn4	8145 X 7183	24843914	14065386	43.384	38.445	35.267	8.266
P143r51_5t19910410_nn5		43468415	26278257	39.546	33.615	29.571	12.030
P143r51_5t19910410_nn6		12195837	10505203	13.862	44.915	41.235	8.193
P143r51_5t19910410_nn7		33544993	18554992	44.686	36.176	32.449	10.302
		181969879	**109721138**	**39.703**	**38.665**	**35.247**	**8.840**

P140r46_5t19880208_nn1		14793591	10608198	28.29	42.065	39.639	5.76
P140r46_5t19880208_nn2		13350037	10127580	24.16	43.495	37.394	14.02
P140r46_5t19880208_nn3		21234274	12859193	39.44	39.864	36.615	8.150
P140r46_5t19880208_nn4	8178 X 7151	22309343	13073208	41.40	39.264	36.036	8.221
P140r46_5t19880208_nn5		35437610	21362439	39.71	35.418	31.302	11.621
P140r46_5t19880208_nn6		11308640	10254401	9.32	47.351	44.571	5.871
P140r46_5t19880208_nn7		26388369	25408107	41.61	38.161	34.446	9.735
		144821864	**93690126**	**35.306**	**40.802**	**37.143**	**8.967**
P144r47_5t19891121_nn1		13872181	10274014	25.938	42.924	40.701	5.178
P144r47_5t19891121_nn2		11778466	9605687	18.447	44.576	42.689	4.233
P144r47_5t19891121_nn3		18816517	11785964	37.363	41.407	38.666	6.619
P144r47_5t19891121_nn4	7846 X 7541	23465669	13434090	42.750	39.503	36.253	8.227
P144r47_5t19891121_nn5		36037459	20180377	44.001	36.158	32.144	12.096
P144r47_5t19891121_nn6		10426827	9889660	5.151	45.715	41.893	8.360
P144r47_5t19891121_nn7		24729780	14079133	43.068	39.462	36.008	8.752
		139126899	**89248925**	**35.850**	**41.392**	**38.336**	**7.383**
P144r45_5t19891105_nn1		12983325	9892694	23.804	43.225	41.154	4.791
P144r45_5t19891105_nn2		11400907	9381796	17.710	44.673	42.888	3.995
P144r45_5t19891105_nn3		17921821	11343323	36.706	41.651	39.026	6.302
P144r45_5t19891105_nn4	7847 X 7542	23216508	13125777	43.463	39.505	36.259	8.216
P144r45_5t19891105_nn5		33533336	18107385	46.001	36.826	32.976	10.454
P144r45_5t19891105_nn6		10791798	9938137	7.910	45.520	41.813	8.143
P144r45_5t19891105_nn7		22611428	13110128	42.019	40.058	36.879	7.935
		132459123	**84899240**	**35.905**	**41.636**	**38.713**	**7.020**
P145r51_5t19910102_nn1		18103143	12332599	31.87	40.092	36.987	7.744
P145r51_5t19910102_nn2		15133043	10855756	28.264	42.196	39.442	6.526
P145r51_5t19910102_nn3		23414160	14099923	39.780	38.802	35.433	8.682
P145r51_5t19910102_nn4	8086 X 7106	27502878	15112472	45.051	37.313	33.998	8.911
P145r51_5t19910102_nn5		39769184	24570078	38.218	34.363	30.417	11.483
P145r51_5t19910102_nn6		12714010	10631922	16.376	44.640	41.259	7.573
P145r51_5t19910102_nn7		28090692	16180854	42.397	37.581	33.976	9.592
		164727110	**103783604**	**36.996**	**274.987**	**251.512**	**8.536**

Confusion matrix [11] is used to assess the accuracy of an image classification. The strength of a confusion matrix is that it identifies the nature of the classification errors, as well as their quantities. In confusion matrix rows correspond to classes in the test set, columns correspond to classes in the classification result. The diagonal elements in the matrix represent the number of correctly classified pixels of each class. The off-diagonal elements represent misclassified pixels. The overall accuracy is calculated as the total number of correctly classified pixels divided by the total number of test pixels.

Another measure which can be extracted from a confusion matrix is the kappa coefficient [10] which is a popular measure to estimate agreement in categorical data. The motivation of this measure is to extract from the correctly classified percentage the actual percentage expected by chance. Thus, this coefficient is calculated as

$$K = \frac{P_0 - P_e}{1 - P_e} \tag{14}$$

P_e is the Expected agreement =

$$\left[\frac{\frac{cm^1 xrm^1}{n} + \frac{cm^2 xrm^2}{n} + \ldots\ldots\ldots \frac{cm^n xrm^n}{n}}{n} \right]$$

cm^1,, cm^2, ……… cm^n are the column 1, 2 …………n marginals

rm^1, rm^2, ……… rm^n are the row 1, 2---------n marginals

n is the total number of test pixels.

The higher the value of kappa, the better the classification performance. If all information classes are correctly identified, kappa takes the value 1. As the off-diagonal entries increase, the value of kappa decreases.

Cross-Validation [12] is a statistical method of evaluating and comparing learning algorithms. The basic form of cross-validation is k-fold cross-validation. In k-fold cross-validation the data is randomly partitioned into *k* equally (or nearly equally) sized segments or folds. Each time, one of the k subsets is used as the training set and the other *k-1* subsets are put together to form a test set. Then the average error across all *k* trials is computed. The advantage of this method is that all observations are used for both training and testing.

For classification two data sets were used. One was a 1040 X 1040 (Mono_lake image)b Landsat TM with all 7 bands. The second data set contained 500 samples of four ground types, barren land, vegetation, forest and rock of the same scene. This second data set is used to observe the classification accuracy. All the 2000 set patterns were classified simultaneously with the Maximum Likelihood, Mahalanobis and Minimum distance classifiers. Classification performance of all the classifiers is displayed in tables 2, 3 & 4. Figure 10 shows the Overall accuracy of all the classifiers. Classification performance by cross validation with different k's is calculated. Tables 5, 6 &7 displays the classification performance with k=4. Figure 11 shows the Overall accuracy of all the classifiers with cross validation of k=4. It is observed that classification performance on proposed compression images is same as JPEG standard compression images and original images. It is also observed that classification performance of Maximum Likelihood classifier> Mahalanobis classifier>Minimum distance classifier.

Table 2 : Confusion Matrix for Maximum Likelihood classifier

Original Image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	100	500	500	0	0	0
2. Vegetation	91	500	0	455	42	3
3. Forest	99.8	500	0	1	499	0
4. Rock	99.8	500	0	1	0	499
Misclassification= 2.35%		Overall accuracy= 97.65%			Kappa coefficient=0.9686	
Standard JPEG Compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.8	500	499	0	0	1
2. Vegetation	89	500	0	445	53	2
3. Forest	99	500	0	5	495	0
4. Rock	100	500	0	0	0	500
Misclassification= 3.05%		Overall accuracy= 96.95%			Kappa coefficient=0.9593	
Proposed compressed image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	100	500	500	0	0	0
2. Vegetation	91.2	500	0	448	36	16
3. Forest	99	500	0	5	490	5
4. Rock	100	500	0	0	02	448
Misclassification= 2.45%		Overall accuracy= 94.33%			Kappa coefficient=0.9244	

Table 3 : Confusion matrix for Mahalanobis classifier Original Image

Original Image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99	500	495	5	0	0
2. Vegetation	96.2	500	0	481	19	0
3. Forest	97.4	500	10	3	487	0
4. Rock	84.8	500	13	63	0	424
Misclassification= 5.65% Overall accuracy= 94.35% Kappa coefficient=0.9246						
Standard JPEG Compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.6	500	498	2	0	0
2. Vegetation	92.6	500	8	463	29	0
3. Forest	95.4	500	11	12	477	0
4. Rock	80.6	500	14	82	1	403
Misclassification= 7.95% Overall accuracy=92.05% Kappa coefficient=0.894						
proposed compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	97	500	485	7	6	2
2. Vegetation	90.2	500	18	451	27	04
3. Forest	91.8	500	14	20	459	7
4. Rock	88.2	500	13	39	7	441
Misclassification= 9.2% Overall accuracy= 91.8% Kappa coefficient=0.8906						

Table 4 : Confusion matrix for Minimum distance classifier

Original Image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.6	500	498	0	0	2
2. Vegetation	77.2	500	0	386	114	0
3. Forest	92.2	500	0	39	461	0
4. Rock	43.4	500	222	60	1	217
Misclassification= 21.9 Overall accuracy= 78.1% Kappa coefficient=0.708						
Standard JPEG compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	98.6	500	493	0	0	7
2. Vegetation	76	500	0	380	119	1
3. Forest	93.6	500	0	29	468	3
4. Rock	43.8	500	235	45	1	219
Misclassification= 22% Overall accuracy= 78% Kappa coefficient=0.706						
proposed compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	97	500	485	0	0	15
2. Vegetation	78	500	0	390	103	7

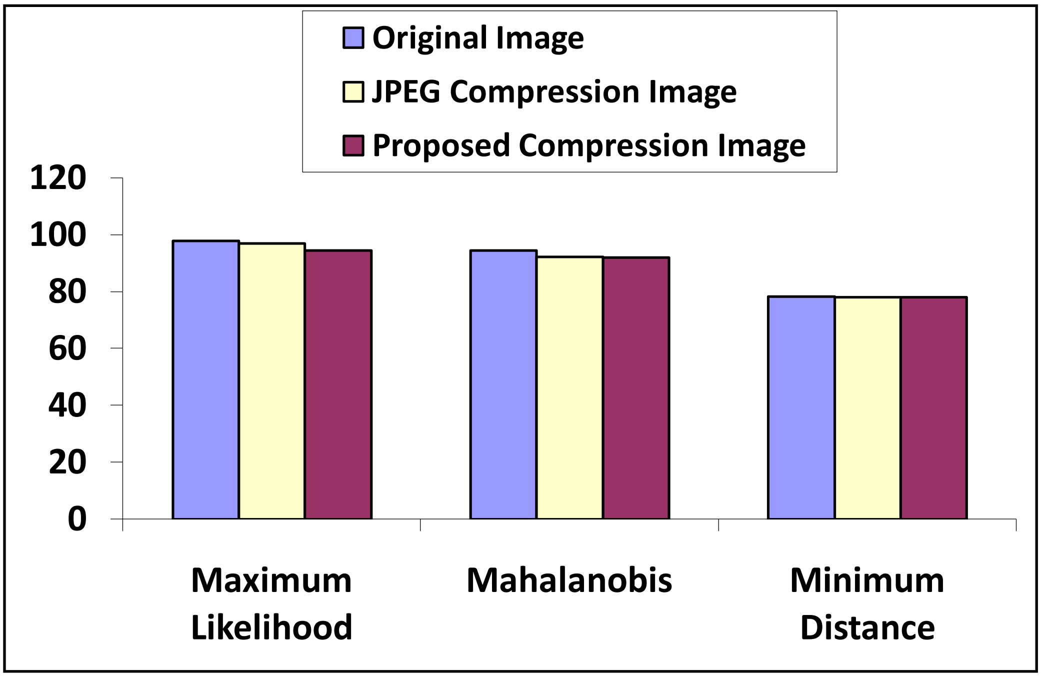

Figure 10 : Overall Accuracy of Classifiers

Table 5 : Average Confusion matrix with cross validation (k=4) of Maximum Likelihood classifier

original image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.86	750	749	1	0	0
2. Vegetation	96.4	750	6	723	20	1
3. Forest	96.8	750	2	20	726	2
4. Rock	92.53	750	20	35	1	694
Misclassification= 3.6%	Overall accuracy=96.4%		Kappa coefficient=0.952			
Standard JPEG compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.6%	750	747	3	0	0
2. Vegetation	96.13%	750	6	721	20	3
3. Forest	96.93%	750	2	19	727	2
4. Rock	92.26%	750	20	35	3	692
Misclassification= 3.77%	Overall accuracy= 96.23%		Kappa coefficient=0.9497			
proposed compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.6%	750	747	3	0	0
2. Vegetation	96%	750	6	720	21	3
3. Forest	96.53%	750	3	19	724	4
4. Rock	92%	750	20	36	4	690
Misclassification =3.97%	Overall accuracy=96.03 %		Kappa coefficient=0.9470			

Table 6 : Average Confusion matrix with cross validation(k=4) for Mahalanobis classifier

original image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.46	750	746	4	0	0
2. Vegetation	94.26	750	12	707	30	1
3. Forest	94.26	750	5	36	707	2
4. Rock	88.80	750	26	54	4	666
Misclassification= 5.8%	Overall accuracy= 94.2%		Kappa coefficient=0.9226			
Standard JPEG compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.60	750	747	3	0	0
2. Vegetation	94.26	750	18	707	25	0
3. Forest	90.93	750	6	54	682	8
4. Rock	86.26	750	31	64	8	647
Misclassification= 7.24%	Overall accuracy= 92.76%		Kappa coefficient=0.9034			
proposed compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.46	750	746	4	0	0
2. Vegetation	93.46	750	21	701	23	5
3. Forest	91.06	750	13	42	683	12
4. Rock	86.8	750	39	46	14	651
Misclassification= 7.3%	Overall accuracy= 92.7%		Kappa coefficient=0.902			

Table 7 : Average Confusion matrix with cross validation (k=4) for Minimum distance classifier

original image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.73	750	748	0	0	2
2. Vegetation	86.13	750	1	646	103	0
3. Forest	67.6	750	0	242	507	1
4. Rock	67.2	750	157	77	12	504
Misclassification= 19.84%	Overall accuracy= 80.16%		Kappa coefficient=0.7354			
standard JPEG compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	99.73	750	748	0	0	2
2. Vegetation	86.13	750	4	646	100	0
3. Forest	67.86	750	0	240	509	10
4. Rock	66.67	750	176	65	9	500
Misclassification= 19.9%	Overall accuracy= 80.1%		Kappa coefficient=0.7346			
proposed compression image						
Spectral Class	Correct Classification (%)	Number of Samples used	Classified as group			
			1	2	3	4
1. barren land	100	750	750	0	0	0
2. Vegetation	86.93	750	3	652	95	0
3. Forest	69.2	750	0	231	519	0
4. Rock	69.2	750	183	43	5	519
Misclassification= 18.67%	Overall accuracy= 81.33%		Kappa coefficient=0.751			

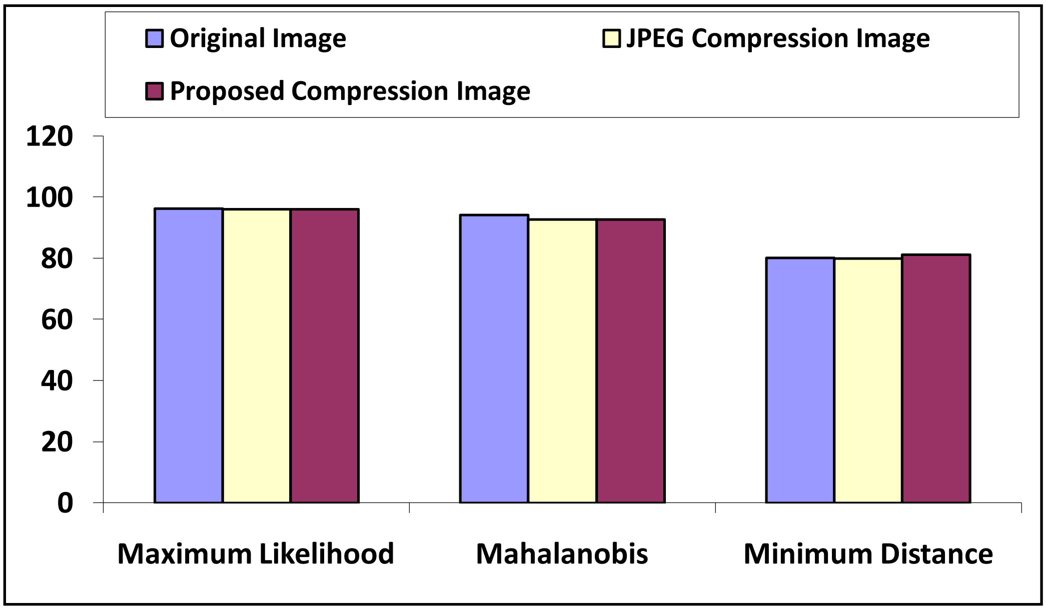

Figure 11 : Overall Accuracy of Classifiers with Cross Validation (K=4)

VI. Conclusions

In this paper, a new Zooming-Shrinking based JPEG compression algorithm is proposed. We have compared our proposed algorithm with Standard JPEG compression. From our experiments it is evident that our approach gives better compression ratios compared to Standard JPEG. The PSNR resulting from our approach is slightly less than Standard JPEG approach. Also the Classification accuracy of original images, Standard JPEG compression images and proposed compression images are almost same.

If a typical satellite mission goal is classification only, then we can send compressed images from satellite which saves bandwidth requirements of a satellite mission. Also, storage requirement reduces by many folds as we will be storing compressed images only. This indirectly reduces power requirement needs of the storage system. In addition, loading and storing of images takes less time compared to original images, thus response times of imaging systems increases.

References Références Referencias

1. Darrel Hankerson, Greg A. Harris, Peter, D. Johnson Jr, "Introduction to information theory and data compression" second edition. A CRC press company.
2. John Watkinson (2004). "MPEG Handbook".
3. Bormin Huang, Antonlo J. Plaza, Joan Serra-Sagrista, Chulhee Lee, Younsong LI, Shen-En QIan (2010, August). "Satellite Data compression, communications and Processing VI". 3-5.
4. John R. Jensen (n.d.). "Introductory Digital Image Processing A Remote Sensing Perspective" Second Edition, Prentice Hall.
5. Cung Nguyen (n.d.). "Detecting Computer–Induced Errors in Remote-Sensing JPEG Compression Algorithms".
6. R.C. Gonzalez and R.E. Woods (1993). "Digital Image Processing", 2nd Edition Addison Wesley.
7. NB Venkateswarlu & PSVSK Raju, (1991). "Three Stage ML Classifier",. *24*, 1113-1116.
8. NB Venkateswarlu & PSVSK Raju. (1994). "Winograd's method: a perspective for some pattern recognition problems". 105-109.
9. NB Venkateswarlu & PSVSK Raju. (1993). "A new fast classifier for remotely sensed imagery,. *International Journal of Remote Sensing, 14*, 383-389.
10. Anthony J. Viera, MD; Joanne M. Garrett, (2005, May). "Understanding Interobserver Agreement: The Kappa Statistic".
11. Morton. J. Canty, "Image Analysis, Classification and Change Detection in Remote Sensing: With Algorithms for ENVI/IDL", A CRC press company.
12. Kim H. Esbensen, "Multivariate Data Analysis-In Practice", Aalborg University, Esbjerg.

This page is intentionally left blank