Compression Techniques for Image Processing Tasks

A. Roman-Gonzalez

***Abstract*—This article aims to present an overview of the different applications of data compression techniques in the image processing filed. Since some time ago, several research groups in the world have been developing various methods based on different data compression techniques to classify, segment, filter and detect digital images fakery. In this sense, it is necessary to analyze and clarify the relationship between different methods and put them into a framework to better understand and better exploit the possibilities that compression provides us respect to image processing techniques; compression becomes a very easy techniques to apply without much technical requirement. In this article we will also see the types of compression and specific image compressors.**

***Keywords*—Data compression, image processing, NCD, Kolmogorov complexity, JPEG, ZIP.**

I. Introduction

THE work presented in this paper is an extension of the work presented in [1]. Actually around the world, we can observe more often and increasingly, that various research groups and research works, using data compression for different applications in the image processing field. In fact, we can see that many authors use data compression techniques for classification and / or segmentation images, filter or denoising image, artifacts detection in images, detecting altered images, etc. The analysis of digital images has a great importance in many fields for which there are many methods, processes and techniques as shown in [2], and if the compression techniques can help more easily to these different purposes, it is a breakthrough.

In [3] and [4] the authors present methods of image classification based on data compression techniques, in the first case using a video compressor such as MPEG4 to texture classification and in the second case is used as general purpose compressor as ZIP compressor and images compressor as JPEG lossless compressor. In general, in [5] the author presents a summary of different application of compression techniques in the classification of different data types. The authors of [6] present a new general method for detecting forged images also based on data compression techniques and rate-distortion analysis, in this case, the author used the JPEG lossy compressor. In [7] the authors apply lossy compression to filter the noise in the images. Also the data compression is used as a technique for artifacts detection in satellite images as shown in [8], [9] and [10] where the authors use both lossy compression and lossless compression.

The use of compression techniques in different areas described above is very interesting and useful; it becomes possible on the basis of information theory, complexity theory and the various applications around these notions.

A. Roman-Gonzalez, Universidad Nacional San Antonio Abad del Cusco, Universidad Andina del Cusco, Universidad Alas Peruanas; Perú. & TELECOM ParisTech – France.
avid.roman-gonzalez@ieee.org

The information theory was developed by Claude E. Shannon in 1948 to find fundamental limits in compression and reliable storage of data communication.

Considering the probability approach, we can establish a first principle of the measurement information. This principle establishes that the message that has more probability, it provide less information. This can be expressed as follows:

$$I(x_i) > I(x_k) \Leftrightarrow P(x_i) < P(x_k) \qquad (1)$$

Where:

$I(x_i)$: Amount of information provided by x_i

$P(x_i)$: Probability of x_i

According to this principle, it is the probability of a message to be sending and not its content, which determines their informational value.

For a message x_1, with an occurrence probability $P(x_1)$, the information content can be expressed by:

$$I(x_i) = \log_2 \frac{1}{P(x_1)} \qquad (2)$$

Where: $I(x_i)$ will have as unit, the bit.

Within information theory, we have the algorithmic approach that talk about the Kolmogorov complexity *K(x)* which is defined as the length of the shortest program capable to producing *x* on a universal machine. Intuitively, *K(x)* is the minimum amount of information necessary to generate *x* through an algorithm.

$$K(x) = \min_{q \in Q_x} |q| \qquad (3)$$

Q_x is the set of codes that instantly generate *x*. The Kolmogorov complexity $K(x|y)$ from *x* to *y* is defined as the length of the shortest program that computes *x* when *y* is given as an auxiliary input for the program. The function $K(x|y)$ is the length of the shortest program that produces the concatenation of *x* and *y*. But *K(x)* is a non-calculable function.

Both, the probabilistic approach of Shannon and the algorithmic approach of Kolmogorov for information theory, have a relationship with data compression.

The compression serves to transport the same information, but using the least amount of space. Data compression is fundamentally based on search data series repetitions. There are two general approaches for compression: lossless compression and lossy compression.

In lossless compression approach, we can mention ZIP compressor, JPEG-LS compressor and Delta compressor. Compressor ZIP is a general purpose lossless compressor based on the combination of LZW code and the Huffman code. The compressor JPEG-LS is a lossless compressor, the algorithm begins with a prediction process, each image pixel value is predicted based on the adjacent pixels, after that, it is necessary to apply an entropy encoder, and get the compressed image. Delta compression is the process of computing a compact and invertible encoding of a target file *T* with respect to a source file *S*.

In lossy compression approach we can mention the JPEG -DCT Compressor, the first step in this compressor is to divide the image into blocks of 8x8 pixels, to each block, the second step is to apply a Discrete Cosine Transform (DCT), after that, to apply a quantifier and finally an entropy encoder for to get the image compressed.

The structure of this paper is as follows: In Section II we present the image classification and image segmentation methods based on compression techniques. Section III presents a method for detecting forged images based on data compression (lossy compression). In Section IV is shown a method where data compression also serves for image denoising. Section V presents different methods for artifacts detection in satellite images. Finally in Section VI we present the conclusions and discussion.

II. Image Classification and Image Segmentation Based on Compression

Based on the Normalized Compression Distance (NCD) presented and defined in [11] and [12] which is one applications of Kolmogorov complexity, in [4] the authors perform image classification.

The Normalized Information Distance (NID) is a similarity measure proportional to the length of the shortest program that represents *x* given *y*, as far as the shortest program that represents *y* given *x*. The calculated distance is normalized, meaning that its value is between 0 and 1, 0 when x and y are totally equal and 1 when the maximum difference between them.

$$NID(x,y)=\frac{\max\{K(x\mid y),K(y\mid x)\}}{\max\{K(x),K(y)\}}=\frac{K(x,y)-\min\{K(x),K(y)\}}{\max\{K(x),K(y)\}} \quad (4)$$

Since the Kolmogorov complexity *K(x)* is a non-calculable function, in [11] the authors approximate *K(x)* with *C(x)* where *C(x)* is the compression factor of *x*. Based on this approach we obtain the Normalized Compression Distance NCD (Normalized Compression Distance).

$$NCD(x,y)=\frac{C(x,y)-\min\{C(x),C(y)\}}{\max\{C(x),C(y)\}} \quad (5)$$

Where *C(x,y)* is an approximation of the Kolmogorov complexity *K(x,y)* and represents the file size by compressing the concatenation of *x* and *y*.

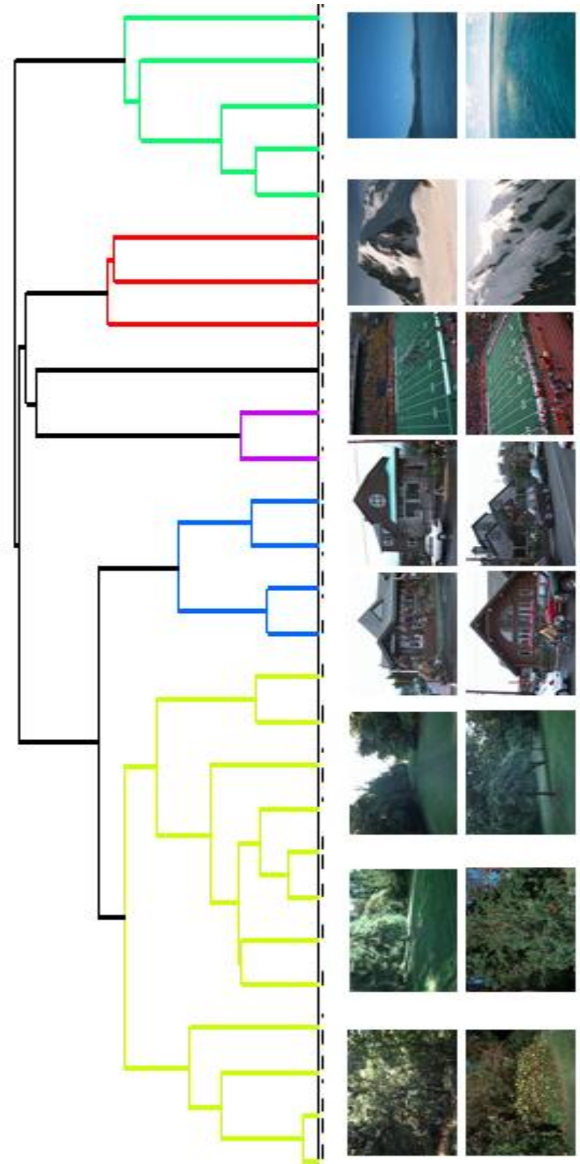


Fig. 1. Image Classification

For image classification using the approach based on Normalized Compression Distance (*NCD*); the first step is calculate the distance matrix between the images based on *NCD* using the Equation 5, thus $d_{ij} = NCD(I_i,I_j)$ where I_i is the *i*-th image of the image database. Thus, we can calculate the distance matrix *D* between all images I_i as:

$$D = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1j} & \dots & d_{1n} \\ d_{21} & d_{22} & \dots & d_{2j} & \dots & d_{2n} \\ . & . & \dots & . & \dots & . \\ d_{i1} & d_{i2} & \dots & d_{ij} & \dots & d_{in} \\ . & . & \dots & . & \dots & . \\ d_{n1} & d_{n2} & \dots & d_{nj} & \dots & d_{nn} \end{bmatrix}$$

The matrix *D* is a square matrix of size *n* x *n*, where *n* is the number of images to classify in the database. Finally we can apply a supervised or non-supervised classification method like SVM, KNN or KMEANS. In this case, we apply a non-supervised classification method like hierarchical classification dendrogram like is shown in Figure 1. The dendrogram is a type of graphical representation of data as a tree that organizes the data into subcategories that are dividing in others to reach the level of detail desired, this type of representation allows appreciating clearly the relationship between data classes. To plot the dendrogram we use the Euclidean Distance method for evaluate the distance between the data using the following instruction in MATLAB:

- Calculate the Euclidean Distance (Distance = pdist(*D*))
- Linkage the distances (Tree = linkage(Distance))
- Read the labels (Labels = importdata('Labels.txt'))
- Plot the dendrogram:

```
dendrogram(Tree,'colorthreshold','default','labels',Labels);
set(Dendrogram,'LineWidth',2)
```

There are also applications of the classification method based on compression techniques, in remote sensing such as those presented in [13] where the authors use the NCD to classify hyperspectral image compressing the spectral signature.

As for classification, we can use the same method described above for image segmentation. The idea is to take an image and divide it into small pieces or patches, we can calculate the distance matrix *D* between these different patches, and then we can classify the patches that will give us as results a segmentation of the original image. This kind of application is frequently for remote sensing applications.

For example, if we take an image *I* of 1024 x 1024 pixels and we divide the image *I* in patches p_i of 64 x 64 pixels, we have 256 patches p_i with i = 1, ……, 256. After applying the NCD to all patches, we obtain de distance matrix *D* composed by $d_{ij} = NCD(p_i,p_j)$ where p_i is the *i*-th patch of the image.

$$D = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1j} & \dots & d_{1n} \\ d_{21} & d_{22} & \dots & d_{2j} & \dots & d_{2n} \\ . & . & \dots & . & \dots & . \\ d_{i1} & d_{i2} & \dots & d_{ij} & \dots & d_{in} \\ . & . & \dots & . & \dots & . \\ d_{n1} & d_{n2} & \dots & d_{nj} & \dots & d_{nn} \end{bmatrix}$$

Finally with this distance matrix *D*, we can apply a supervised or non-supervised classification method like SVM, KNN or KMEANS.

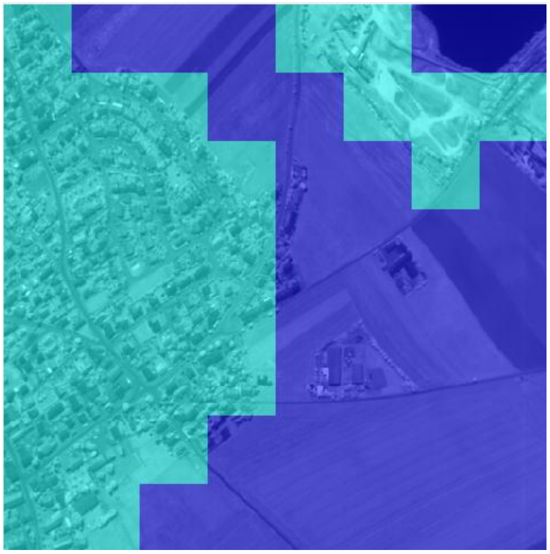


Fig. 2. Image Segmentation

In Figure 2 we can observe an example of the image segmentation based on compression techniques applied to an Earth observation image. We can see clearly the good segmentation between city environment and field environment.

III. Image Fakery Detection

In [6] the authors use the Rate-Distortion experimental curve (RD) using lossy compression, JPEG – DCT lossy compressor. The RD function indirectly measured the visual complexity of the images, for example, plotting the experimental RD curve where the horizontal axis represents the compression factor (size of the compressed image file / size of the original image file) and the vertical axis represents the distortion calculated using the Mean Square Error (MSE); we can make an analysis of the image.

We can say that the falsification of an image can alter the experimental RD curve of the image, for example we can see from Figure 3 that shown in (a) an original image and (b) the same image but with an alteration which the person has been removed.

(a) Original Image [6]

(b) Manipulated Image [6]

Fig. 3: (a) Original photograph of a Car Show. (b) is the same image (a) but the person was erased and replaced by duplication of regions..

Analyzing the image of Figure 3 (a) and (b) using the graph of the experimental RD curves, it can be seen that there is a variation on these curves due to the manipulation of the image. Figure 4 shows the variation, with the blue curve for Figure 3 (b) and the green curve for Figure 3 (a).

Thus, using the variation in the experimental curve RD of images, we can detect when an image was manipulated or not.

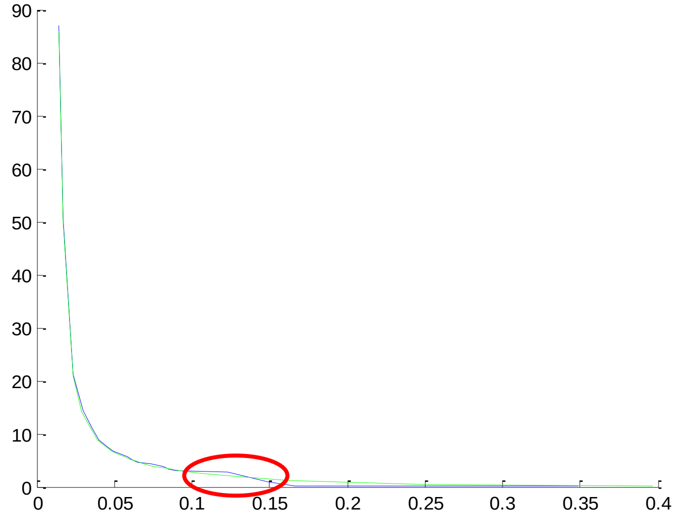


Fig. 4 Rate-Distortion Experimental curve, the horizontal axis represents the compression factor (size of the compressed image file / size of the original image file), the vertical axis represents the distortion calculated using the MSE (mean square error). Blue for Figure 3 (a) and green for Figure 3 (b), the RD experimental curve of the original image is different from the experimental curve RD for image that contains manipulations. [6].

Also it is possible to use the Kolmogorov Structure Function (KSF) for image fakery detection.

In [14] the authors present an analysis about the Kolmogorv's structure function. The relation between an individual data and its model is expressed by Kolmogorov's structure function.

The original Kolmogorov structure function for a data x is defined by:

$$h_x(\alpha) = \min_S \{\log|S| : S \ni x, K(S) \leq \alpha\} \tag{6}$$

Where: S is a contemplated model for *x*.

α is a non-negative integer value bounding the complexity of the contemplated *S*.

The Kolmogorov's structure function $h_x(\alpha)$ tell us all stochastic properties of data *x*. [14].

In the same way, it is introduced the Best-Fit function, given by:

$$\beta_x(\alpha) = \min_S \{\partial(x \mid S) : S \ni x, K(S) \leq \alpha\} \tag{7}$$

Where: *x* is regarded as a typical member of *S*.

The MDL function is given by:

$$\lambda_x(\alpha) = \min_S \{\Lambda(S) : S \ni x, K(S) \leq \alpha\} \tag{8}$$

$$\Lambda(S) = \log|S| + K(S) \geq K(x) - O(1) \quad (9)$$

Where: $\Lambda(S)$ is the total length of two parts of x code with an S model.

The conditional Kolmogorov structure function is given by:

$$h_x(i \mid y) = \min_S \{\log|S| : S \ni x,\ K(S \mid y) \leq i\} \quad (10)$$

The Kolmogorov structure function is a non-computable since the Kolmogorov complexity is also a non-computable function; that is the reason why we use the compression factor as an approximation to complexity.

The KSF is an approximation of the Rate-Function Distortion using Kolmogorov complexity theory. In that sense, we will use this function to see the changes in the experimental curve of the KSF when the image is manipulated, we will use the same test images were used for analysis with the RD curve. In Figure 5 we can observe the KSF curves for images in Figure 3, the blue curve for the image (a) and the green curve for image (b).

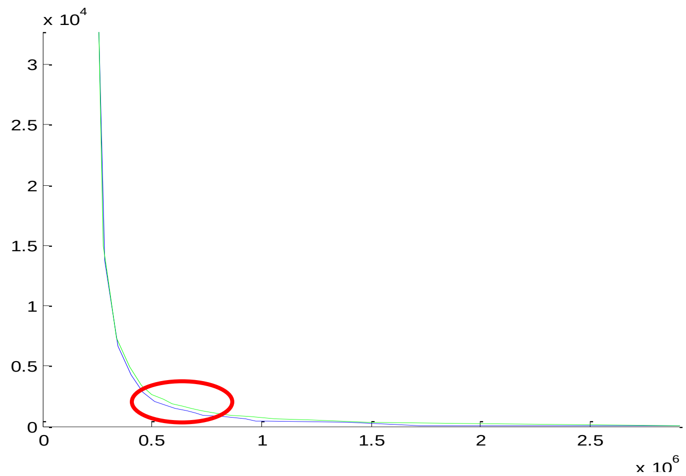


Fig. 5 Experimental KSF curve, the horizontal axis represents the Kolmogorov complexity approximation as the size of the image compressed file in bytes, the vertical axis represents the amount of bits that is needed to represent a model of original image. Blue for Figure 3 (a) and green for Figure 3 (b), the KSF experimental curve of the original image is different from the experimental curve KSF for image that contains manipulations. [6]

It can be seen that as RD curves, in KSF curves, variation may be observed when there is a manipulation of images, we use this variation to determine whether an image has been altered or not.

IV. Image Denoising

A method for filtering and denoising is presented in [7] where the authors use lossy compression techniques for computing experimental RD curve of an image while calculating the Minimum Description Length function [7] and thus obtain the minimum point and identify the image whose noise was eliminated by lossy compression.

Similarly, we can use a new calculation of experimental RD curve as a measure of distortion using the Normalized Compression Distance NCD between the error E (produced by the difference between the original image X and the compressed-decompressed image Y) and the compressed-decompressed image Y. The *Rate-NCD(E,Y)* curve between the error E and the compressed-decompressed image Y should have a particular behavior as shown in Figure 6, for a moment, when there is not much compression, the compressed-decompressed image Y is almost equal to the original image X and hence the error E is little information which makes the distance between E and Y large, this distance decreases as the compression increases, because the information is going to error and this will seem a bit to the image, but there comes a point which (called here *ARG* point) should be more information on the error E that on the compressed-decompressed image Y thus the distance between them will increase again.

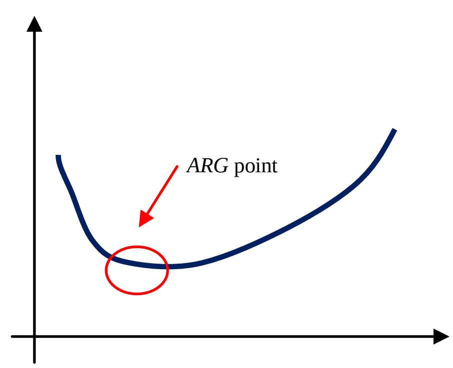


Fig. 6 *Rate-NCD* curve, hypothetical comparison between the error E and the compressed-decompressed image Y.

In this way we can identify the minimum point (*ARG* point) which would correspond to the image without noise as shown in Figure 7 (using the same image used in [7]). This process can also serve to indicate how to find the minimum amount of information needed to interpret the information, how much detail we can loss without to arrive until a not interpretable data. In Figure 7, we can see clearly that the mouse image for the minimum point has the enough detail to recognize a mouse.

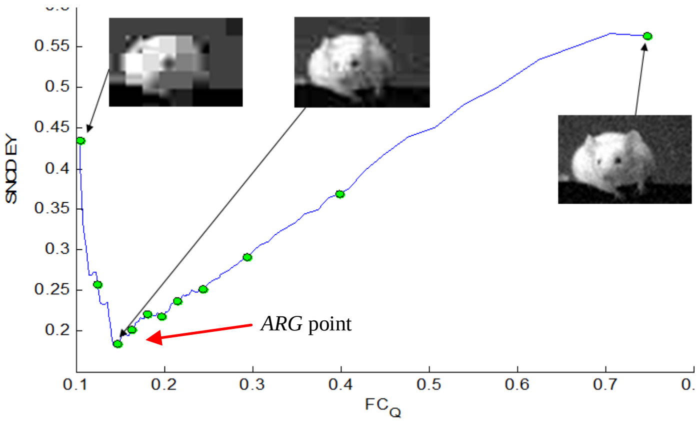


Fig. 7 Image Denoising

V. Artifact Detection

In the research works [8], [9] y [10], the authors present different methods for artifacts detection in satellite images, different methods based on compression techniques, lossless compression and lossy compression. The method that has better results is the method that uses a lossy compression to calculate the experimental RD curve presented in [10]. The idea is to examine how an artifact can have a high degree of regularity or irregularity for compression [10] and analyze the error space produced by the lossy compression. The RD analysis was done as the blocks diagram shown in Figure 8.

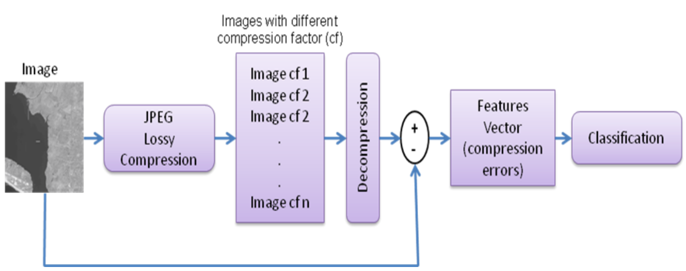


Fig. 8 Block Diagram for Rate-Distortion Analysis [10]

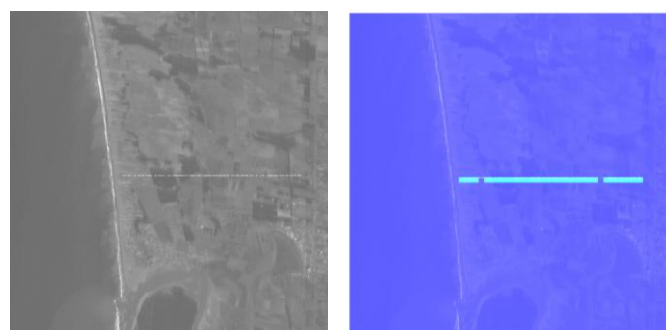
First the author take the image under test I, they dive the image I in different n patches X_i of 64 x 64 pixels, with $i = 1, 2, \ldots, n$, thus $I = \{X_1, X_2, \ldots X_n\}$. For each *i-th* patch X_i, they compress the patch with different quality q using a lossy compression, using different quality they obtain different compression factor. After that, they decompress the image and obtain a decompressed image Y_{iq}. The next step is to calculate the error for each compression factor between the original patch X_i and the compressed-decompressed patch Y_{iq}, for calculate the error; they use the Mean Square Error (MSE). Based on the errors for each compression factor q and for each patch X_i, they compose a features vector $V_i = [F_{i1}, F_{i2}, \ldots F_{iq} \ldots F_{iQ}]$ where $F_{iq} = MSE(X_i, Y_{iq})$. Thus, they have a matrix:

$$V = \begin{bmatrix} F_{11} & F_{12} & \ldots & F_{1q} & \ldots & F_{1Q} \\ F_{21} & F_{22} & \ldots & F_{2q} & \ldots & F_{2Q} \\ . & . & \ldots & . & \ldots & . \\ F_{i1} & F_{i2} & \ldots & F_{iq} & \ldots & F_{iQ} \\ . & . & \ldots & . & \ldots & . \\ F_{n1} & F_{n2} & \ldots & F_{nq} & \ldots & F_{nQ} \end{bmatrix}$$

Finally, with this matrix V, they apply a supervised or non-supervised classification method like *SVM*, *KNN* or *KMEANS*. In this case, the authors use *KMEANS* classification method, which is an unsupervised algorithm; *KMEANS* classifies data according to K (positive and integer value) centroids or groups as assigned, the algorithm calculates the minimum distance of these centroids to all other data and the group as the minimum distance. For our purpose, we take a value of $K = 2$ that represents one group for images with artifacts and other group for images without artifacts.

For authors, the quality factor vary between 0 and 100, so $Q = 101$, and for an image of 512 x 512 pixels, we obtain 64 patches of 64 x 64 pixels, thus n would be $n = 64$.

An example of the application of this method based on the RD analysis is the Dropout detection shown in Figure 9, In this case it is a SPOT image containing actual artifacts, and the detection is done correctly.



(a) © CNES (b)

Fig. 9 Dropout (SPOT). (a) Some electronic losses during the image formation process create these randomly saturated pixels. The dropouts often follow a line pattern (corresponding to the structure of the SPOT sensor). (b) Artifact is detected.

Another method for artifacts detection is based on the similarity space using the Normalized Compression Distance (*NCD*), the authors propose to analysis the possible similarity between images with artifacts due to possible similarity pattern between artifacts. The first step is to take the satellite image I and divide it into n patches X_i of 64x64 pixels like is shown in Figure 10, with $i = 1, 2, \ldots, n$, thus $I = \{X_1, X_2, \ldots X_n\}$. With these patches X_i, after that, it is necessary to calculate the distance matrix between them using the *NCD*, thus we have $d_{ij} = NCD(X_i, X_j)$. Thus, it is possible to calculate the distance matrix D between all patches X_i as:

$$D = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1j} & \dots & d_{1n} \\ d_{21} & d_{22} & \dots & d_{2j} & \dots & d_{2n} \\ . & . & \dots & . & \dots & . \\ d_{i1} & d_{i2} & \dots & d_{ij} & \dots & d_{in} \\ . & . & \dots & . & \dots & . \\ d_{n1} & d_{n2} & \dots & d_{nj} & \dots & d_{nn} \end{bmatrix}$$

The matrix D is a square matrix of size n x n, where n is the number of patches; for an image of 512 x 512 pixels, we obtain 64 patches of 64 x 64 pixels, thus n would be $n = 64$.

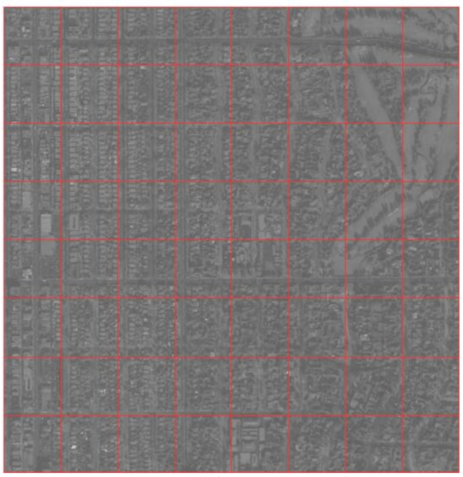


Fig. 10 Separate image in patches of 64x64 pixels for calculate the distance matrix between the patches

Finally, with this distance matrix D, the authors apply a non-supervised classification method like hierarchical classification dendrogram like is shown in Figure 12. The dendrogram is a type of graphical representation of data as a tree that organizes the data into subcategories that are dividing in others to reach the level of detail desired, this type of representation allows appreciating clearly the relationship between data classes.

The blocks diagram and process for application of this method based on *NCD* to artifacts detection, is represented in Figure 11.

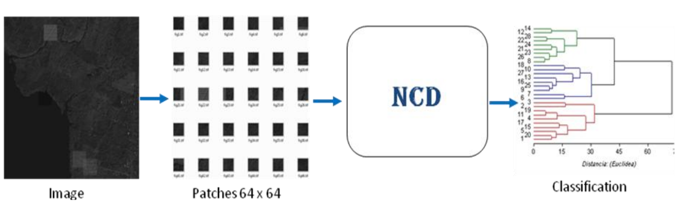


Fig. 11 Experiment Process: we take the satellite image and divide it into patches of 64x64 pixels, with these patches we calculate the distance matrix between them using NCD and finally we applied a hierarchical classification method to cluster and identify the patches with artifacts

We use the NCD to evaluate how patches with artifacts can have a similar structure and see if it is possible to make a cluster with all patches with artifacts.

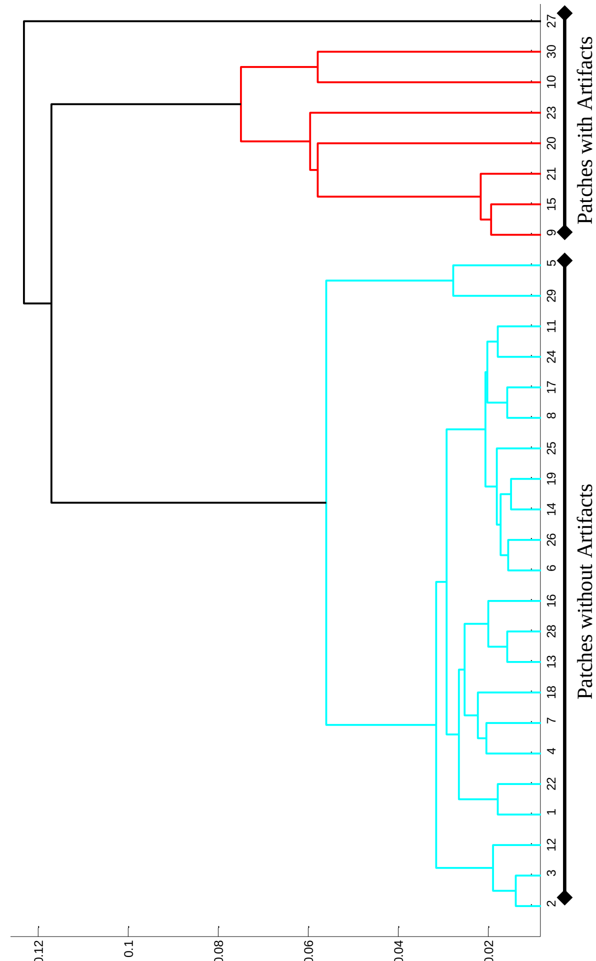


Fig. 12 Hierarchical Classification of the Patches: we can see two cluster, red cluster for patches with artifacts, and cyan cluster for patches without artifacts.

In this dendrogram representation, we can observe clearly how the patches with artifacts and the patches without artifacts can form two different clusters in certain hierarchical level.

VI. Conclusions and Discussion

As has been seen, there are many studies which use data compression techniques for image processing purposes.

In this paper we have tried to present the different developments made by numerous research groups around the world about the applications of data compression, specifically in image processing. Each group uses different techniques, different types of compression methods and different approaches for different applications but all of them related to two important points: the compression and its application to image analysis in order to make this task easier, understandable and with good results.

The use of compression techniques in image processing, makes this task easier and more accessible, since everybody know to use a compressor, everybody use a compressors in your daily lives to reduce the size of the files and to transmit or transport them more efficiently. No specific knowledge is needed to use a compressor.

Through this article, we provide the background and fundaments to better understand why we can use data compression on digital image analysis and thus continue to develop more applications and improving existing ones.

It was clearly observed that the application of data compression for image processing is entirely feasible. It is possible because data compression is fully related to the information theory in the two approaches. In the image processing and image analyzing, the important thing is the image information content, based on this information we can reach a correct analysis.

The results obtained in the different applications are very encouraging, so it is necessary to continue in this line of research and develop more applications.

VII. References

[1] A. Roman-Gonzalez, K. Asalde-Alvarez, "Image Processing by Compression: An Overview", World Congress on Engineering and Computer Science – WCECS 2012; San Francisco – USA; October 2012; pp. 650-654.

[2] A. Roman-Gonzalez, "Digital Images Analysis", Revista ECIPeru, vol. 9, N° 1, 2012, pp. 61-68.

[3] B.J.L. Campana y E.J. Keogh, "A Compression Based Distance Measure for Texture", University of California, Riverside, EEUU 2010.

[4] M. R. Quispe-Ayala, K. Asalde-Alvarez, A. Roman-Gonzalez, "Image Classification Using Data Compression Techniques"; 2010 IEEE 26th Convention of Electrical and Electronics Engineers in Israel – IEEEI 2010; Eilat – Israel; November 2010, pp. 349-353.

[5] A. Roman-Gonzalez, "Clasificación de Datos Basado en Compresión", Revista ECIPeru, vol. 9, N° 1, 2012, pp. 69-74.

[6] A. Roman-Gonzalez, C.J. Reynaga-Cardenas, "Implementacion de un Método General para la Detección de Imágenes Alteradas Utilizando Técnicas de Compresion", Engineering Thesis, Universidad Andina del Cusco, 2012.

[7] S. Rooij, P. Vitanyi, "Approximating Rate-Distortion Graphs of individual Data: Experiments in Lossy Compression and Denoising", IEEE Transaction on Computers, vol.61, N° 3, March 2012, pp. 395-407.

[8] A. Roman-Gonzalez, M. Datcu, "Satellite Image Artifacts Detection Based on Complexity distortion Theory", IEEE International Geosciences and Remote Sensing Symposium – IGARSS 2011, Vancouver – Canada, July 2011, pp. 1437-1440.

[9] A. Roman-Gonzalez, M. Datcu, "Data Cleaning: Approaches for Earth Observation Image Information Mining", ESA-JRC-EUSC Image Information Mining: Geospatial Intelligence from Earth Observation Conference, Ispra – Italy, March 30 – April 1, 2010, pp. 117-120.

[10] A. Roman-Gonzalez, M. Datcu, "Parameter Free Image Artifacts Detection: A Compression Based Approach", 2010 SPIE Remote Sensing, vol. 7830, 783008, Toulouse – France, September 2010.

[11] M. Li and P. Vitányi, "The Similarity Metric", IEEE Transaction on Information Theory, vol. 50, N° 12, 2004, pp. 3250-3264.

[12] R. Cilibrasi, P. M. B. Vitanyi; "Clustering by Compression", *IEEE Transaction on Information Theory*, vol. 51, N° 4, April 2005, pp 1523 - 1545.

[13] A. Roman-Gonzalez, M. A. Veganzones, M. Graña, M. Datcu, "A Novel Data Compression for Remote Sensing data Mining", ESA-JRC-EUSC Image Information Mining: Geospatial Intelligence from Earth Observation Conference, Ispra – Italy, March 30 – April 1, 2010, pp. 101-104.

[14] N. K. Vereshchagin, P. M. B. Vitanyi; (2004, diciembre); "Kolmogorov's Structure Functions and Model Selection"; IEEE Transaction on Information Theory, Vol. 50, N° 12, pp. 3265-3290.

[15] F. Tussell, "Complejidad Estocástica", San Sebastián 1996.

[16] D. McKay, "Information Theory, Inference, and Learning Algorithms", Cambridge University Press, 2003.