MIN-MAX COMPRESSION METHODS FOR MEDICAL IMAGE DATABASES

Kosmas Karadimitriou, and John M. Tyler

Department of Computer Science
Louisiana State University
Baton Rouge, Louisiana 70803
kosmas@acm.org, tyler@bit.csc.lsu.edu

ABSTRACT

The volume of medical imaging data produced per year is rapidly increasing, overtaxing the capabilities of Picture Archival and Communication (PACS) systems. Image compression methods can lessen the problem by encoding digital images into more space-efficient forms. Image compression is achieved by reducing redundancy in the imaging data. Existing methods reduce redundancy in *individual* images. However, these methods ignore an additional source of redundancy, which is based on the common information stored in more than one image in a *set of similar images.* We use the term "*set redundancy*" to describe this type of redundancy. Medical image databases contain large sets of similar images, therefore they also contain significant amounts of set redundancy.

This paper presents two methods that extract set redundancy from medical imaging data: the Min-Max Differential (MMD), and the Min-Max Predictive (MMP) methods. These methods can improve compression of standard image compression techniques for sets of medical images. Our tests compressing CT brain scans have shown an average of as much as 129% improvement for Huffman encoding, 93% for Arithmetic Coding, and 37% for Lempel-Ziv compression when they are combined with Min-Max methods. Both MMD and MMP are based on reversible operations, hence they provide *lossless* compression.

I. INTRODUCTION

Medical imaging is a field that has experienced significant advances due to new computer technologies. Digital systems have become an integral part of CT, MRI, PET, SPECT, and Ultrasound imaging and even traditionally non-digital techniques (e.g. film X-rays) are gradually evolving into computerized imaging. However, digital imaging requires storing, communicating and manipulating large amounts of digital data. Studies have shown that the radiology department of a large hospital can produce more than 20 terabits of image data per year [1]. This is the result of the high image resolution used in radiology, and the large number of images required for each examination. For example, a CT exam produces an average of 12 Mbytes of imaging data (30 images, 512x512 pixels each, 12 bits/pixel), whereas Digital Subtraction Angiography produces 21 Mbytes of imaging data (15 images/exam, 1024x1024 pixels each, 8 bits/pixel) [2]. In addition to the inherently digital modalities, digital data are also generated by digitizing X-ray images. A digitized chest X-ray is 4096x4096 pixels, 12 bits/pixel, which requires 24 Mbytes of storage space per image.

Overall, the amount of digital radiologic data generated every year in the USA alone is on the order of petabytes (10^{15}) and is increasing rapidly [2]. This stretches the capabilities of digital storage systems, and imposes exceedingly high requirements on the bandwidth of communication networks [3,4].

Digital image compression can address these problems by reducing the data storage and transmission requirements. Many compression methods have been developed and have been evaluated for the medical environment [2]. These compression methods usually reduce the size of the data 2-3 times with no information loss, and more than 10 times with some information loss.

Despite the higher compression ratios of lossy compression methods, their use in medical imaging is limited because of concerns on losing image details [5,6]. Even when the compression is visually lossless, an unsuccessful diagnosis from an image that has lost some information may lead to legal implications [2]. Another reason for avoiding lossy compression is the development of computer-aided diagnosis techniques. Computerized analysis of an image can use even the smallest details (e.g., very smooth variations in pixel intensities) which are often invisible to the eye. Compression methods should not lose any of these potentially important image details. For this reason, in medical imaging lossless compression is more important than lossy compression. The two methods presented in this paper are both *lossless* compression methods.

In general, lossless compression can be achieved by taking advantage of data redundancies. Existing methods can efficiently reduce data redundancies in individual images [7]. However, medical image databases contain an additional type of redundancy, the "set redundancy" [8], which can be used to achieve even higher compression ratios. In the next section, we will examine this type of redundancy, and then we will present methods that can exploit it for better compression.

II. SET REDUNDANCY

A. The Concept of "Set Redundancy"

As stated above, interpixel, psychovisual, and coding redundancy are the three types of redundancy found in still monochrome images. However, in a set of *similar images*, one can observe that there exists a significant additional amount of *inter-image* redundancy. "Similar images" are images that have:

a) similar pixel intensities in the same areas
b) comparable histograms
c) similar edge distributions
d) analogous distributions of features

According to this definition, medical images that are produced by the same modality and depict a particular organ or part of the human body (e.g., brain CT scans) are "*similar*" to each other. For example, consider a set of 500 CT brain scans. There is a statistical correlation between the images in this set, and every one of these images contains some information already stored in the other images. This creates an inter-image redundancy, the "*set redundancy*", which we define as follows:

<u>Definition</u>: ***Set redundancy*** is the inter-image redundancy that exists in a set of similar images, and refers to the common information found in more than one image in the set.

Set redundancy can be used to improve compression. A limit to compression is imposed by the image entropy. In the next section we will show how set redundancy can be used to decrease the average image entropy in a set of similar images.

B. Image Entropy and Set Redundancy

The concept of *entropy* is defined in information theory as a measure of information [9]. The *image entropy* measures the amount of information an image contains, and it is also used as a measure of the compressibility of the image (lower entropy means better compressibility). The entropy of an individual image is calculated by using the histogram distribution of its pixel values, which represent "individual image statistics". However, in a set of images, "set statistics" can be used instead of "individual image statistics", resulting in smaller average image entropy. The reason is that in sets of similar images every pixel position [x,y] is associated with its own histogram distribution of gray values. By the definition of similarity among images (similar values at similar positions), these distributions will be highly non-uniform. If these distributions are used to encode the pixel values, then the entropy will be very small. In the following, we will study theoretically the decrease of image entropy due to the use of "set statistics" and then we will present two methods to implement these ideas in practice.

Consider a set of similar images with S% similarity among the images. For example, if S = 0.4 then, on the average, for every pixel position, 40% of the pixel values across all images will have the same value. Note that some areas of the image may have higher variability than others; however, *on the average* the similarity will be 40%. In every pixel position there is variability V% among the images.

$$\text{Similarity}: S$$
$$\text{Variability}: V = 1.00 - S$$

Suppose that there are $(n+1)$ symbols in the alphabet $\{a_0, a_1, a_2, \ldots, a_n\}$ and a_k is the symbol with the highest frequency S:

$$P(a_k) = S$$

For simplicity, assume that the other symbols appear with equal probability, V/n

$$P(a_0) = \ldots = P(a_{k-1}) = P(a_{k+1}) = \ldots = P(a_n) = V/n$$

so that,

$$P(a_0) + P(a_1) + \ldots + P(a_n) = S/100 + n\,V/(100\,n) = S + V = 1.0$$

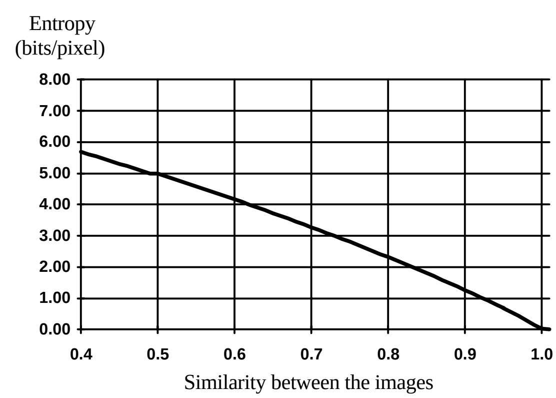
Then, the entropy H is

$$H = -S\log(S) - \underbrace{(V/n)\log(V/n) - \ldots - (V/n)\log(V/n)}_{n \text{ times}}$$

or,

$$H = -S\log(S) - V\log(V/n) \qquad (1)$$

where $n = 255$ for 8-bit gray-scale images. Figure 1 presents the values of entropy H resulting from (1) for different values of S. The entropy clearly decreases as the similarity increases among the images in the set.



<u>Fig. 1:</u> Entropy vs. similarity in sets of similar images

Based on “individual image statistics”, the *first-order* entropy for 8-bit gray-scale natural images is usually 5-7 bits/pixel [7]. However, figure 1 shows that the first-order entropy estimate can be smaller than this when “set statistics” are used. When similarity is more than 40%, the average entropy of the images in the set is less than 6 bits/pixel. For 80% similarity the entropy is 2.32 bits/pixel, and for 90% similarity it is 1.27 bits/pixel. If higher-order entropy estimates are used, then these numbers will be even smaller. This clearly suggests that for sets of images with high similarity, it is advantageous to use set statistics instead of individual image statistics in compressing the images. Implicitly, the use of set statistics reduces the set redundancy from the images, resulting in better compression ratios.

The entropy can be further reduced if set statistics are coupled with individual image statistics. In this case, each image in the set can be compressed using the “locally optimal” distribution; in other words, the statistics that produce the best compression.

The problem with the above schemes is that they require storage of a histogram table for every pixel position. For images with 8 bits per pixel, this is a 256-entry table. Clearly, it is not practical to keep so much statistical data for every pixel position. A practical method must reduce set redundancy using only limited set statistics. The Min-Max Differential method [10] is a method that reduces set redundancy using only the minimum and maximum values instead of the whole histograms. Section III(A) will review the Min-Max Differential (MMD) method and in section III(B) the new Min-Max Predicted (MMP) method will be presented which is based on predictive coding using the minimum and maximum values.

III. IMAGE COMPRESSION BASED ON SET REDUNDANCY

In order to incorporate the concept of set redundancy reduction for compressing sets of similar images, a two-step procedure can be used (figure 2). In the first step, the images are decorrelated from the set by extracting the set redundancy; in the second step, the images are compressed by using any compression method. Note that set redundancy extraction is a completely independent step, therefore there are no restrictions about which compression method to be used in the second step.

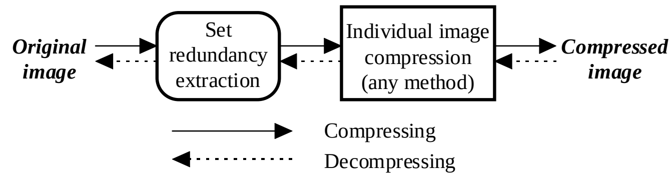


Fig. 2: Coupling set redundancy extraction with image compression

Ideally, set redundancy extraction must satisfy the following requirements:

(a) it must reduce (ideally, eliminate) the set redundancy from all images in the set
(b) it must not be computationally expensive because its computations are in addition to the computations required in the compression of the second step
(c) it must enable the compression and decompression of individual images from the set, without requiring global calculations on the whole set
(d) it must be lossless

Min-Max Differential encoding (MMD) is a method that satisfies all the requirements for practical set redundancy extraction. Specifically, it reduces set redundancy with only a small amount of calculations, it is easy to implement, it compresses or decompresses individual images from the set, and it is lossless. The next section will review this method, and in section B another new method will be presented, the Min-Max Predictive (MMP) method.

A. The “Min-Max Differential” (MMD) Method

The MMD method stores statistical information from a set of similar images in the form of a “min” and a “max” image. To create the “min” image, MMD compares for every pixel position the pixel values across all images, and chooses the smallest. Similarly, the “max” image is created by selecting the largest pixel value for every pixel position. Then, MMD processes every image in the set by replacing the original pixel values with the differences from either the “min” or the “max” image (whichever is smaller). This reversible operation reduces the dynamic range of pixel values, so that any standard compression method can be used on the MMD-processed images with improved results.

Figure 3 presents graphically the MMD operation. The abscissa represents the pixel positions and the ordinate the pixel values. Every curve describes an image. The curves for the “min”, the “max”, and a random image from the set are depicted. The difference values that replace the original image values are shown as dotted lines. Note that the differences are calculated from either the “min” or the “max” curves, depending on which one yields the smaller difference value. When the difference is larger than (*max*-*min*)/2, MMD switches to the other curve. In this way the decoder is synchronized with the encoder, while the smallest possible difference values are selected and used.

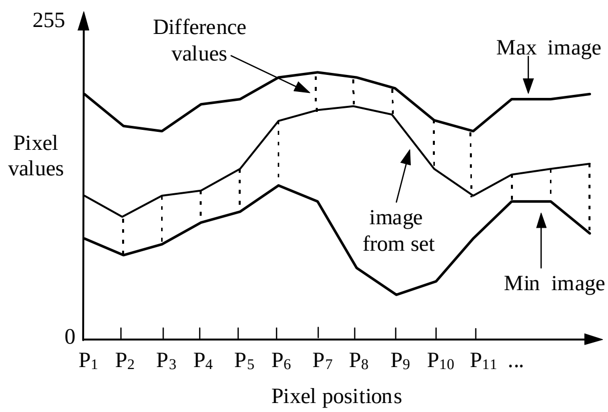


Fig. 3: The Min-Max Differential method

B. The “Min-Max Predictive” (MMP) Method

The MMP method also uses “min” and “max” images. For every pixel position P_i, the “min” image gives the minimum value min_i across all images in the set. Similarly, the “max” image provides the maximum value max_i. These two values are the limits on the range of possible values that pixel P_i may assume. Furthermore, neighboring pixels tend to fall in approximately the same area between the minimum and the maximum values (figure 4). By dividing the range between the minimum and maximum values into N levels, we can represent the position of every pixel between its corresponding minimum and maximum values as a “level” L_i

$$L_i = N\left(\frac{P_i - \min_i}{\max_i - \min_i}\right)$$

Neighboring pixels usually have approximately the same “level”, even though their actual values may differ considerably. For example, consider the following values:

Max value	192	205	211	197	183	190	199	204
Pixel value	71	78	89	85	69	80	87	95
Min value	54	60	71	68	51	64	70	78

For every pixel position, the range between its maximum and minimum values can be divided into N=256 “levels”, where 0 represents the minimum value, and 255 the maximum value. According to this definition, the above pixel values correspond to the following “levels”:

Pixel value	71	78	89	85	69	80	87	95
Level	32	32	33	34	34	33	33	34

As this example demonstrates, the “level” values have smaller variation than the pixel values. Therefore the “levels” are better predictors for the next pixel values than the pixel values themselves. This observation led to the development of the MMP method. The MMP method predicts the value of a pixel P_i by using the “level” information from its previous pixel P_{i-1}. Pixel P_{i-1} and its min_{i-1} and max_{i-1} are used to determine its level L_{i-1}. This can be used directly as a predictor for the level of the current pixel ($L_i = L_{i-1}$). A variation with slightly better results is to set $L_i = (L_{upper} + L_{left}) / 2$ where L_{upper} is the level of the upper neighbor pixel and L_{left} the level of the left neighbor. L_i , min_i and max_i are used to calculate a predicted value for pixel P_i. The difference between this predicted value and the original value is stored in the image replacing the original value. This concludes the encoding process. To recover the original image from its “difference image”, the decoding process follows the same steps to calculate the predicted values, and then it adds the difference values obtained from the “difference image”.

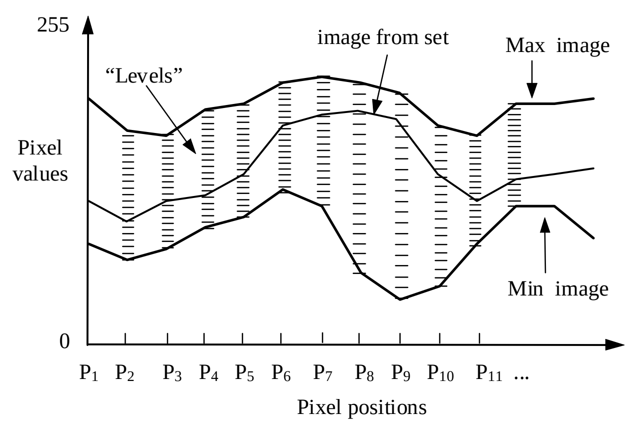
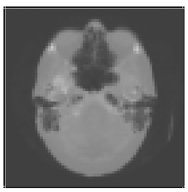
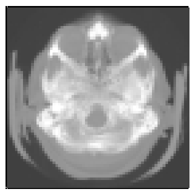


Fig. 4: The Min-Max Predictive method (N=20)

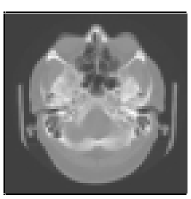
IV. EXPERIMENTAL RESULTS

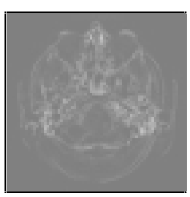
A test database of 51 brain CT images was used to evaluate the performance of our methods. All these images were obtained at MD Anderson Cancer Center in Houston, Texas. Each of these images contained 512x512 pixels, 8 bits/pixel, and were randomly selected from different patients. From this test database, a set of images with high set redundancy was formed using the ten most similar images, and the “min” and “max” images were created for that set:

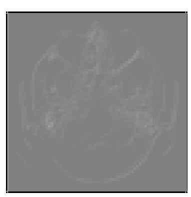

“min” image


“max” image

Note that although registration in a standard position and size helps to create the “min” and “max” images, there is no need to store the registered versions of the original images. Once the registration parameters (rotation, translation, and scaling) are known, the “min” and “max” images can be registered directly on the original images using these parameters. Then the proposed methods can be used to extract set redundancy. An example of the effect of MMD- and MMP-processing on one of the original images is the following :


Original image


MMD-processed


MMP-processed

As the above example shows, both MMD and MMP-processed images have a significantly smaller range of gray-scale values than the original image. In principle, this leads to improved compression ratios for any compression method. To measure the improvement in compression, we used three different compression methods that are widely used in practice. These are the Huffman [11], Arithmetic [12], and Lempel-Ziv compression [13]. Huffman compression was implemented with the Unix command *pack*. Lempel-Ziv compression was implemented with the Unix command *compress*, which is the LZW version [14] of the Lempel-Ziv method. For Arithmetic coding, the publicly available software based on [12] was used.

To test the MMD method, all images were compressed by the three compression methods mentioned, with and without using MMD pre-processing. The results are presented in Table 1. As this table shows, MMD method improved Huffman compression by 48%, Arithmetic compression by 28%, and Lempel-Ziv compression by 13%. These improvements are due to the extraction of set redundancy from the images, before individual image compression removes other types of redundancies.

The MMP method was tested in the same way. All images were compressed using the three test compression methods, with and without MMP pre-processing. The results are presented in Table 3. It is clear that MMP, which is more sophisticated than MMD, resulted in higher improvement on all three methods: 129% for Huffman, 93% for Arithmetic, and 37% for Lempel-Ziv method. MMP performed better than MMD because for every pixel position it uses <u>both</u> the *min* and *max* values, whereas MMD uses <u>either</u> the *min* <u>or</u> *max*. The minor increase in implementation complexity for MMP is easily justified by the payoff of better compression. Both methods required less than 2 seconds of computation time on a Sun SPARC 20 computer to compress or decompress an image file of 262 Kbytes.

<u>Table 1:</u> Experimental results from implementing the MMD method on the CT images

Compression technique	**Average lossless compression**	**Compression improvement**
Huffman MMD + Huffman	1.379 : 1 2.047 : 1	 +48 %
Arithmetic MMD + Arithmetic	1.701 : 1 2.183 : 1	 +28 %
Lempel-Ziv MMD + Lempel-Ziv	2.449 : 1 2.758 : 1	 +13 %

<u>Table 2:</u> Experimental results from implementing the MMP method on the CT images

Compression technique	**Average lossless compression**	**Compression improvement**
Huffman MMP + Huffman	1.379 : 1 3.157 : 1	 +129 %
Arithmetic MMD + Arithmetic	1.701 : 1 3.275 : 1	 +93%
Lempel-Ziv MMD + Lempel-Ziv	2.449 : 1 3.359 : 1	 +37 %

V. CONCLUSION

As radiology becomes increasingly digital, the amount of digital imaging data for storage or transmission grows rapidly. This imposes a serious problem to the realization of an all-digital radiologic environment. Efficient compression methods help alleviate this problem by reducing the size of imaging data. After more than half century of research in data compression, many good compression methods have emerged. However, all of these methods have been designed to compress individual images, rather than sets of images.

Compression is based on the elimination of data redundancies. Sets of similar images contain a special type of redundancy, the "*set redundancy*", which is defined as the common information found in more than one image of the set. Medical image databases usually store large sets of similar images, and therefore contain large amounts of set redundancy. We developed two new methods, the Min-Max Differential (MMD) and Min-Max Predictive (MMP) methods that increase compression in sets of similar images by targeting set redundancy. Both methods can be combined with any compression method, because set redundancy extraction is independent of individual image compression. Tests with three widely used lossless compression techniques, Huffman, Arithmetic, and Lempel-Ziv compression, have shown compression improvement of as much as 129% when set redundancy is extracted. Both MMD and MMP methods are fast, lossless, and easy to implement. Future research will investigate the combination of set redundancy extraction with lossy compression techniques, as well as ways to integrate set redundancy extraction into medical image database systems.

ACKNOWLEDGEMENT

The medical images used in our compression experiments were provided by M.D. Anderson Cancer Center, Houston, TX, USA, and part of this research was performed at the Radiologic-Pathologic Institute of M.D. Anderson Cancer Center. The authors would like to thank Dr. Sidney Wallace, Dr. Marc Fenstermacher, and John Grossman for their efforts in making the images available to us and for their support on this project.

REFERENCES

[1] A.W. Wong, R.K. Taira, and H.K. Huang, "Implementation of a digital archive system for a radiology department", in *Proc. SPIE Conf. on Medical Imaging VI: PACS Design and Evaluation* 1645, pp. 182-190, 1992.

[2] S. Wong, L. Zaremba, D. Gooden, and H.K. Huang, "Radiologic Image Compression - A Review", *Proc. IEEE*, vol. 83, no. 2, pp. 194-218, Feb. 1995.

[3] W.J. Chimiak, "The digital radiology environment", *IEEE Journal on Selected Areas in Commun.*, vol. 10, no. 7, pp. 1133-1144, Sep. 1992.

[4] J.R. Cox, E. Muka, J.G. Blaine, S.M. Moore, and G.R. Jost, "Considerations in Moving Electronic Radiography into Routine Use", *IEEE Journal on Selected Areas in Commun.*, vol. 10, no. 7, pp. 1108-1120, Sep. 1992.

[5] T. Ishigaki, S. Sakuma, M. Ikeda, Y. Itoh, M. Suzuki, and S. Iwai, "Clinical evaluation of irreversible image compression: Analysis of chest imaging with computer radiography", *Radiology*, vol. 175, no. 3, pp. 739-743, June 1990.

[6] J.M. Bramble, "Comparison of information-preserving and information losing data-compression algorithms for CT images", *Radiology*, vol. 170, pp.453-455, Feb. 1989.

[7] M. Rabbani and P.W. Jones, *Digital Image Compression Techniques*, SPIE Press, 1991.

[8] K. Karadimitriou, *Set Redundancy, the Enhanced Compression Model, and Methods for Compressing Sets of Similar Images*, Ph.D. Dissertation, Louisiana State University, Baton Rouge, LA, Aug. 1996.

[9] C.E. Shannon, "A mathematical theory of communication", *Bell Syst. Tech. J.*, vol. 27, pp. 379-423, July 1948.

[10] K. Karadimitriou and J.M. Tyler, "The Min-Max Differential Method for Large-Scale Storage and Compression of Medical Images", in *Proc. of Annual Molecular Biology and Biotechnology Conference*, Baton Rouge, Feb. 1996.

[11] D.A. Huffman, "A method for the construction of minimum redundancy codes", *Proc. IRE*, no. 40, pp. 1098-1101, Sep. 1952.

[12] A. Moffat, R. Neal, I.H. Witten, "Arithmetic Coding Revisited", *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 1995.

[13] J. Ziv, and A. Lempel, "A universal algorithm for sequential data compression", *IEEE Trans. Inf. Theory*, vol. IT-23, no. 3, pp. 337-343, May 1977.

[14] T.A. Welch, "A Technique for High Performance Data Compression", *IEEE Computer*, vol. 17, no. 6, pp. 8-19, June 1984.