ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20180407110315
Original scientific paper

Software System for Efficient Lossless Compression of Medical Images

Josip KNEZOVIĆ, Martin ŽAGAR, Hrvoje MLINARIĆ

Abstract: In medical science and surgery preparation, latest trends such as 3D and 4D bio-modeling, telesurgery, telepresence and imaging data archival produce a great amount of data. To achieve a satisfying level of speed in data manipulation and using less storage it is necessary to use some compression method. Lossy compression techniques generally achieve better compression, but reconstructed image differs from the original. Medical imaging is specific regarding compression methods– some important parts of data, such as the region of interests should be stored and transmitted in a lossless way. That is why lossless compression should be employed for those vital parts of interest for diagnostic and surgery analysis purposes. Previous research showed that predictive coding techniques are very effective in lossless compression. Therefore, we propose a complete software system for medical image compression, decoding, and viewing based on our predictive, lossless image compression method CBPC 1. We also introduce new features in the algorithm in order to make it more practical by reducing its computational complexity, while at the same time not incurring its compression efficiency. These improvements substantially improve the processing speed and make our proposed software suitable for integration into current and future paperless hospital information systems. Our software was extensively tested against the compression efficiency and computing time as shown in this paper proving its deployment in medical applications where images need to be delivered with minimum delay using limited communication throughput.

Keywords: lossless compression software; medical image compression; telemedicine; 3D CAS telesurgery; transmission

1 INTRODUCTION

On a yearly basis, based on diagnostic devices such as CT and MRI, the overall volume of imaging data in medical institutions can reach several terabytes, directly affecting the way how the data is stored and the amount of storage capabilities [1, 2]. That is why it is really important to adopt efficient compression techniques and to inspect all possible redundancies in data, both in spectral, spatial and temporal domain to achieve cost reduction. This is especially important for the upcoming era of ubiquitous, embedded, portable and low power computing devices [3-5]. From the perspective of compression efficiency, lossy compression can achieve better compression results, based on the fact that higher frequencies, regions that are not similar when matching to some kernel shapes or even whole frames that have just minimal distortion can be reduced or completely removed in the encoding process. On the other hand, lossless compression techniques in medical applications should be used because even a minimal distortion in the data should be encoded and not be lost. Medical imaging data compression enables efficienton-line transmission and availability of patient diagnostic imaging data which is essential for future mobile and ubiquitous electronic health frameworks [6-8].

In this paper, we develop a new method for lossless image compression based on highly adaptive prediction-based technique CBPC [9, 10] that we previously reported [11]. Our previous technique is characterized by efficient lossless compression with the drawback of high computational requirements and therefore impractical execution time. In order to make it more practical, we revised it with an effective heuristic optimization described in this paper that strictly follows statistical characteristics of medical image data. This optimization helped us develop a complete software tool for medical image compression, decompression and viewing with practical execution time and throughput. We also carefully approached the revision of our previously proposed compression algorithm in order to make it more suitable for integration into the modern hospital software and information systems. Our efforts resulted in the SCBPC method. The optimizations we have employed resulted in comparative compression efficiency to our previous proposal with halved execution time, making it useful for practical uses in modern hospital information and image archival systems.

2 COLLABORATIVE WORK

Results and algorithms developed in this work present continuation of our work in this field, together with the medical experts from Clinics for Tumors in Zagreb [11]. Conducted research and findings on the user and expert experience on using lossless compression in the field of medical imaging resulted in the extension of our lossless method and overall software system that we present here. Before the introduction of digital medical imaging, recordings were stored mostly on films which resulted in the inevitable loss of information obtained from the diagnostic device. Scanning the film images and storing them on the computer systems also has its drawbacks because of losing parts of significant data. Subsequent image manipulations were practically impossible. Storing the digital images as they are generated by diagnostic devices was finally enabled by the introduction of medical imaging protocols and standards such as DICOM and NIfTI [12, 13]. These images can be properly explored, manipulated or inspected by the use of powerful computer and information systems, either manually or semi-automatically with the help of an expert [14]. This is of special importance in cases where complex and thorough examinations are used, or in cases with preoperative preparations which require rapid and precise demarcation between the disease involved and intact tissues. In the new era of information processing, techniques mentioned above present the basis for new trends in developing spatial or volumetric medical models as surgical aids in each, the preparation, procedures and postoperative analysis. Nowadays, computerized imaging enables the simulation of entire operative procedures using complex spatial models and simulated operative field entry [14, 15].

3 MATERIALS AND METHODS

Our approach adapts previously designed and developed compression method CBPC [11]. The starting

point of this proposed compression framework is to diminish the computational complexity and on the other hand to preserve compression efficiency. This is enabled with our innovative approach based on heuristics in the initial step of the compression method which results in the prediction of the method. This could be also tested through the graphical user interface (GUI) which we developed for our compression method, making it a self-contained standalone application.

Adaptation of CBPC is based on the optimization of the CBP predictor due to the fact that the algorithm showed that changes in the predictor are mostly detected on the boundaries of different regions of interest [16]. This is resulting in an adaption of predictor coefficients which is specific to our approach because usually once these coefficients are calculated, they remain mostly unchanged as long as the coder is in that specific region of interest and specific dominant property (for example some smooth region). On the other hand, when coder moves to some other previously detected region of interest, CBP predictor is changed significantly because of that new dominant property changes to the new region of interest. In the next step of the algorithm, this is propagated in the edge areas. The novelty of our approach is based on this change, so we are able to reduce the number of iterations in computing the CBP predictor by not changing the CBP coefficients as long as the examined pixel is located in the current region of interest. Implementation aims towards finding when the magnitude of the prediction error on the current pixel is beyond a predefined threshold, in order to perform the complex computational calculations of the CBP predictor weights. If the prediction error magnitude is below the threshold, previously calculated predictor coefficients are not changed and used for the current pixel. With using this method, we avoided pixel-by-pixel based computation of CBP predictor coefficients and calculating prediction penalties for pixels beyond a predefined threshold.

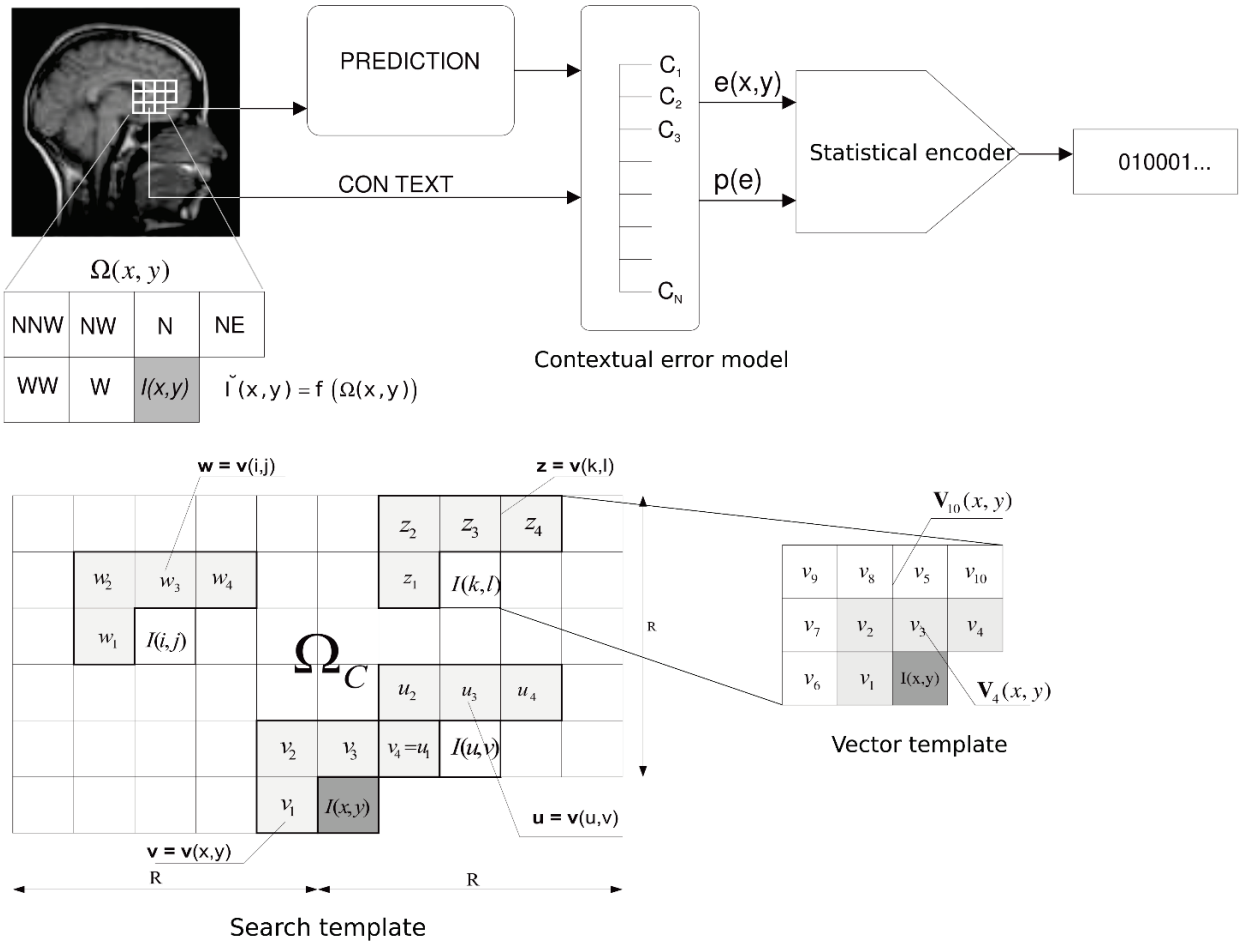

Figure 1 Elements of proposed prediction scheme

The proposed predictor is based on the idea of blending predictors from [17], which is adapted to the dynamic classification of predictor context based on regions of interest, on a pixel-by-pixel basis. The initial set of predictors $F = \{f_1, f_2, \ldots, f_N\}$ consists of N static predictors defined in order to describe the presence of the specific dominant property of the region of interest. Tab. 1 provides description of the initial set of static predictors used in our algorithm. As it is normally used, in our work we denote geographical orientation for notation of surrounding pixels where N is, for example, upper (north) pixel from the current pixel, i.e. *WW* indicates pixel with the distance of two pixels left (west) of the current pixel. We propose seven basic predictors denoted in Tab. 1, so we can denote both vertical and horizontal edge predictors as well as planar predictors. The blending process of these predictors results with the construction of an averaging predictor in order to predict well in the noisy regions. Having this defined, final computed predictor will be able to adapt to most relevant image properties while detecting and

adjusting to the specific dominant properties of specific regions of interests, whether it is an edge, a planar region or even a texture with the nontrivial shape.

Table 1 Set of static predictors

Static predictor	Prediction function	Description
f_1	N	Upper pixel to the current pixel
f_2	W	Left pixel to the current pixel
f_3	NW	Up left pixel to the current pixel
f_4	NE	Up right pixel to the current pixel
f_5	$N + W - NW$	Planar region predictor
f_6	$2N - NN$	Planar region predictor horizontal
f_7	$2W - WW$	Planar region predictor vertical

Our predictive method differs from traditional, adaptation-based predictors in its specific classification phase which selects the neighboring pixels on which the computation of predictor parameters will be performed. Unlike previous approaches which statically select a set of closest previously coded pixels [18], we define a larger search window and dynamically select a set of predictors based on the estimation of image properties and their similarity to the current unknown pixel surrounding. The classification process ends with setting the neighboring pixels on which the blending of the final predictor F is performed. This is based on the outcomes of [19] in order to be usable in symmetric, backward adaptive algorithm similar.

Fig. 1 shows our system model for CBP prediction. Ω_C denotes the causal context used by the predictor which we will refer to as a search template, shown on the bottom left part of Fig. 1. It is important that the region which is examined is the window of size R composed of previosuly encoded pixels on which the search procedure for classification is performed. We connect each pixel from Ω_C and current, unknown pixel $I(x, y)$ with its vector template $\boldsymbol{v}(x, y)$ shown bottom right. Vector template is composed of d closest causal neighboring pixels. In Fig. 1 we present the vectors of size $d = 4 : \boldsymbol{v}_4(x, y)$ and $d = 10 : \boldsymbol{v}_{10}(x, y)$. On the other side of search template we depict the vectors of size four composing of four closest neighbors, depicted as N, W, NW, and NE pixels. With using of the Euclidian distance between associated vectors, we enable the classification of pixels into arrays of similar elements. In order to reduce the computational complexity, we use some simplifications. Based on [16] we start with calculating only the current array with current pixel being examined, following the array size (pixels in that array) is set to some constant M at the beginning of the coding process. In reality, this could be changed, but we preserve the array size.

Assuming that the image exhibits local stationary property, we can freely assume that the currently calculated prediction function will efficiently predict the value of all the unknown pixels in the current local region. This means that recalculation of the prediction coefficients for the next pixel is unnecessary as long as the pixel belongs to the region. This idea is implemented in our new proposal with the following heuristic modification of the original CBP predictor: If the previously calculated predictor predicts well for the current pixel, then it will be subsequently used for the following pixels, meaning that the compute-intensive re-computation of predictor weights is skipped. This omission of predictor recalculation is obtained all the way until the pixel for which the current predictor results in the prediction error above some predefined threshold value. At this moment, we guess that we have reached the boundary of the local region and stepped into a new local region with different statistical properties, meaning that it is the time to completely recalculate or reset our adaptive prediction function.

The idea above is implemented into the CBP predictor and integrated into the complete lossless compression scheme that we call Selective Classification and Blending Predictive Coder or SCBPC. The complete coder operates through the following steps:

Parameters: Pp - previous prediction error, T - predictor re-computation threshold, M - classification cell size, R - search window radius, d - vector size, F - set of static predictor:

1. **Step - initialization**: in the initialization step, if any previous prediction error exists, it is reset.
2. **Step - algorithm iteration**: calculate pixel-based iteration with the following procedure:
3. **Step - selective computation**: calculate the previous prediction error, if it is less than the prediction threshold T, then go to step 4, otherwise, go to step 5.
4. **Step - classification**: For each pixel $I(i, j) \in \Omega_C$ compute the Euclidian distance $D(i, j)$ between its corresponding vector $\boldsymbol{v}(i, j)$ and the current pixel's vector $\boldsymbol{v}(x, y)$:

$$D(i,j) = \| \boldsymbol{v}(i,j) - \boldsymbol{v}(x,y) \| = \sum_{k=1}^{d} | v_k(i,j) - v_k(x,y) |^2 \qquad (1)$$

where d is the size of the vector template, and v_k's are vector components as shown in Fig. 1. Based on the computed distances, determine M pixels from Ω_C that belong to the current cell, i.e. with minimal vector distances from the current pixel's vector.

The current cell will be used as a set of pixels on which the blending of the predictors from F will be performed. We denote those pixels as blending context Ω_B for the set F.

5. **Step - blending**: On the dynamically selected blending context Ω_B perform the blending of the set $F = \{f_1, f_2, \ldots, f_N\}$ of static predictors. For every static predictor f_k the penalty G_k is calculated by the following equation:

$$G_k = \frac{1}{M} \sum_{I(i,j) \in \Omega_B} \left(\hat{I}_k(i,j) - I(i,j) \right)^2 \qquad (2)$$

where $\hat{I}_k(i,j) = f_k(i,j)$ is the predicted value of predictor f_k for the pixel $I(i, j)$. Based on the penalties we form the prediction for the current pixel $\hat{I}(x, y)$ as:

$$\hat{I}(x,y) = f_P(x,y) = \left[\frac{\left(\sum_{k=1}^{N} \frac{1}{G_k} \cdot \hat{I}_k(x,y) \right)}{\sum_{k=1}^{N} \left(\frac{1}{G_k} \right)} \right] \qquad (3)$$

The output of this step gives the predictor of the current pixel. This is calculated as a weighted sum of predictions of all the static predictors from F with weights inversely proportional to corresponding differences in computations caused by the blending context, which is reversed proportional to efficiency. If the predictor predicts well, its contribution in the final prediction will be higher. Thus it has more chance to produce a precise current prediction. In other way, it will be penalized because of huge computational differences and the importance will be diminished (blended).

6. **Step - error correction:** On the blending context Ω_B calculate typical error of the final predictor as:

$$\overline{e}(x,y) = \frac{1}{M} \sum_{I(i,j)\in\Omega_B} \left(f_P(i,j) - I(i,j) \right) \tag{4}$$

Based on the computational differences and blending process, current pixel is calculated with adding weight (denominator) to reduce the noise and is defined as:

$$I_P(x,y) = \hat{I}(x,y) + \overline{e}(x,y) \tag{5}$$

This final step of proposed predictor captures typical bias of the blending predictor f_p on the set of pixels with similar properties as the current pixel. Based on the current pixel prediction calculate the current prediction error.

7. **Step - contextual error modeling:** Forward the current pixel value and the current prediction error to the contextual error model. Calculate the contextual state for the current pixel and map the error value to a statistical symbol.
8. **Step - statistical coding:** Given the contextual state and symbol, statistically encode the symbol to produce the coding bit stream. Go to step 3.

By performing this algorithm CBP predictor is adjusted to the dominant property of the region of interest and other non-dominant properties are also included by blended, but still minimum weighted predictors. This inclusion makes critical advantage to the predictors that cannot be re-modeled and cannot have more than one property focusing on, while our solution is providing a more precise prediction. In the case that one of the penalties is zero, the corresponding predictor is used for prediction without any other additional steps. This is resulting in a prediction based on the perfect predictor (with no error) and this predictor will be used for prediction of next pixel as well. The other case is representing predictor that has some error (and penalty) so the predictor will be calculated as an average with using the blending process and noise reduction in the final calculation which is considered as a great advantage in order to reduce the noise in the image [19].

Fig. 2 presents the steps of computations on a test medical data. The left part is the original image while on the right side, white pixels denote the positions for which the recomputation of the prediction function was performed. Initially, the threshold T was set to zero which resulted in a reduction of recomputation pixels by a factor of 2.05 and reduced the computation time to 39% of the original CBP predictor. On the other hand, the compression efficiency was just slightly incurred by approximately 2% compared to our original CBPC method. We have also examined a sort of variable threshold algorithm in which the threshold was not statically chosen but determined based on the absolute value of the previous error that initiated the predictor recomputation. However, the gains obtained were diminishing compared to the added complexity of the algorithm. The threshold step was included in the method in order to reduce computation time in the first place, so we omitted any additional complexity step from our proposal.

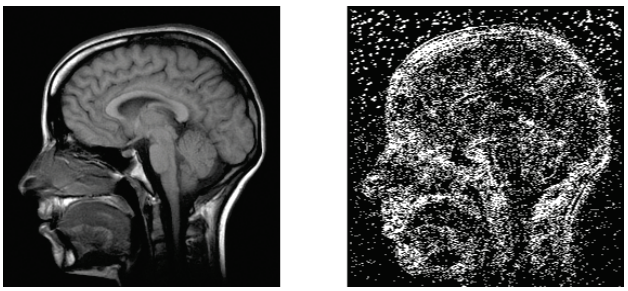

Figure 2 Effects of the selective computation of prediction function (Te=0)

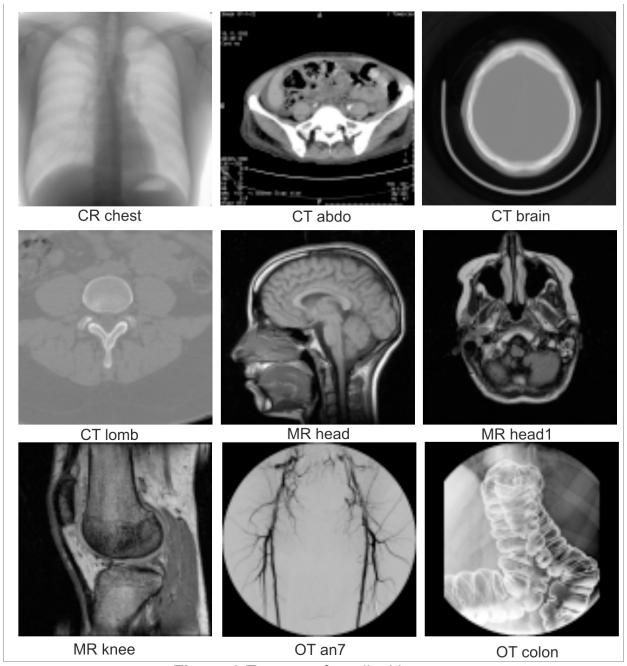

Figure 3 Test set of medical images

4 COMPARATIVE EVALUATION

Evaluation of our proposed method will be focused on setting the benchmark for the performance of proposed predictors based on the set of nine grey (8-bit) medical images obtained with different modalities [20] from Fig. 3. Tab. 2 shows experimentally chosen working parameters of the CBPC1 and SCBPC methods used in our evaluation.

Table 2 Working parameters of encoders

Predictor	R	M	d	T
CBPC	5	7	4	-
SCBPC	6	7	4	0

R – radius, M – cell size, d – vector size, T – threshold

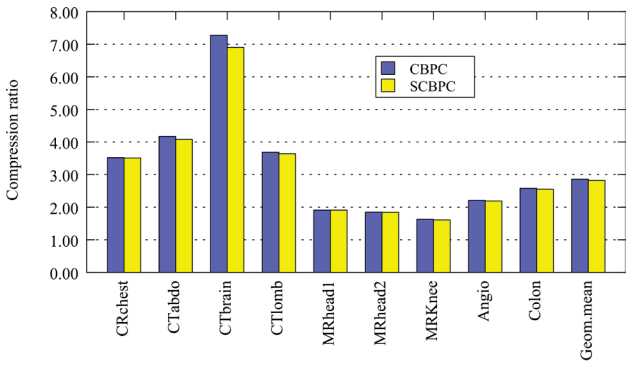

Figure 4 Compression efficiency comparison between CBPC and SCBPC

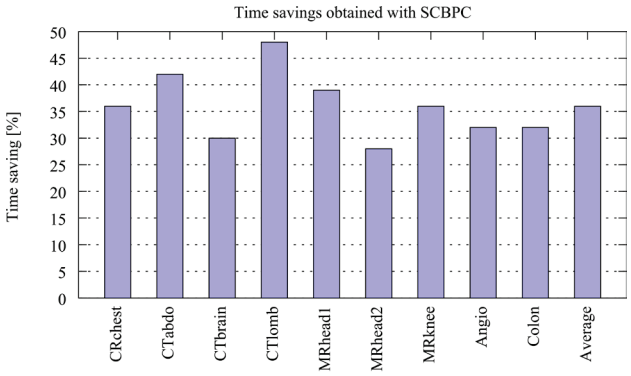

Figure 5 Time savings obtained with SCBPC compression method

Table 3 Compression ratios and time savings obtained on the test images

		CR chest	CT abdo	CT brain	CT lomb	MR head1	MR head2	MR knee	Angio	Colon
Compression ratio	CBPC	3.51	2.94	6.84	3.65	1.76	1.77	1.56	1.92	2.13
	SCBPC	3.49	2.91	6.72	3.65	1.72	1.76	1.55	1.91	2.12
Time saving (%)		36	42	30	48	39	28	27	32	32

Tab. 3 illustrates the effects of selective computation of predictor weights on the compression efficiency and on the execution time. We show the compression ratio of the predictive part of CBPC and SCBPC encoder which encompass all the steps except contextual error modeling (6) and statistical encoding (7). We performed this test in order to examine the efficiency of predictors solely. As shown in Tab. 2, for the SCBPC we set the threshold T to be zero. Therefore, the SCBPC predictor will be recomputed as soon as it generates the prediction error with the absolute value strictly greater than zero (1, 2, and so on). In terms of the compression efficiency, SCBPC predictor performs slightly worse than CBPC predictor, while, on average, we obtained processing time reduction by 36%. This clearly demonstrates the benefits of our proposal making a compute-intensive CBPC predictor more practical in real use through its adaptation to SCBPC predictor.

We also compared the complete coders (CBPC and SCBPC) that incorporate arithmetic coding in the statistical encoding step (7). Fig. 4 shows their compression efficiency by reporting the compression ratios obtained on the test set. SCBPC method closely follows the CBPC encoder maintaining reduced computational complexity. As for the compression efficiency, the geometric mean of compression ratios obtained with SCBPC encoder is within 2% of the CBPC results. Fig. 5 indicates the time savings of SCBPC encoder compared to CBPC encoder for our benchmark images showing that average execution time is reduced by more than 35%.

Table 4 Comparison with other compression methods

Image	CALIC	JPEG-LS	JPEG 2000_R	CBPC	SCBPC
CR chest	3.40	3.35	3.17	3.52	3.51
CT abdo	3.52	4.23	3.09	4.17	4.08
CT brain	6.45	6.20	5.63	7.27	6.90
CT lomb	3.62	3.42	3.36	3.69	3.64
MR head	1.87	1.80	1.79	1.91	1.91
MR head1	1.80	1.73	1.70	1.85	1.84
MR knee	1.61	1.57	1.57	1.63	1.61
OT an7	2.16	2.18	2.02	2.21	2.19
OT colon	2.49	2.50	2.32	2.58	2.55
Geom. Mean	2.72	2.71	2.52	2.86	2.82

Tab. 4 compares the system outcomes for CBPC and SCBPC coders and compares the results with other lossless coders for our benchmark. There are some well-known algorithms such as CALIC algorithm [21], and JPEG 2000 which is contrary to other JPEG algorithms, lossless and it is based on wavelet transform [22]. Last two columns show the results of our previously proposed CBPC encoder and SCBPC encoders proposed in this work. The average performance of our proposed methods is better than the standard algorithms defined in our benchmark.

Finally, we have developed a GUI graphical user interface fronted for CBPC and SCBPC compression methods. The only difference to the user in choosing SCBPC over CBPC is in the selection of selective computation in the predictor and in the setting of the recomputation threshold value (defaulted to zero). However, even with default, highly restrictive threshold, SCBPC performs faster than CBPC while maintaining similar compression efficiency as demonstrated previously in our experiments. This is very attractive in cases where huge batches of medical images are compressed in order to be archived or transmitted over a resource-constrained media. Additionally, the GUI interface allows setting of basic method parameters or more advanced tweaking, image compression, decompression and viewing on the computer screen. Furthermore, we extended the application in order to allow easy image inspection, manipulation, transcoding and archival. The complete software package can be easily integrated into existing and future hospital information systems.

5 CONCLUSIONS

Our proposed system for efficient lossless compression of medical images provides significant spare in the sense of data needed to be sent via telecommunication canal and memory needed to store it, up to 33%. Our results are generated from the benchmark images we set before but giving the more than satisfying results. Our future work will focus on adapting our system for compression on other medical data, for example, to enable virtual reality in medical surgeries. In the context of virtual surgery, our innovative approach considers using of

some additional data to visual overlay over existing 3D MRI or CT data, in order to obtain blending of interactive digital elements – like selecting regions of interests– into our real-world environments [23]. This could be enabled by using the results from our research which covers 3D surface analysis and classification techniques, to make it possible to selectively propose portions of medical volumetric data which consists of regions of interest and which are sent to our extended 3D lossless compression method based on SCBPC. Data that are outside of our region of interests (for example backgrounds), which are medically irrelevant could be compressed with some lossy method making the method meet the requirements of precise medical needed for different aspects of virtual diagnostics and virtual surgeries.

6 REFERENCES

[1] Kharyuk, P. V., Oseledets, I. V., & Ushakov, V. L. (2014). Compression of fMRI data using wavelet tensor train decomposition. *Vych. meth. Programming, 15*(4), 669-676.

[2] Al-Khafaji, G. & George, L. E. (2013). Fast Lossless Compression of Medical Images based on Polynomial. *International Journal of Computer Applications, 70*(15), 28-32. https://doi.org/10.5120/12039-7999

[3] Rad, R. M., Attar, A., & Shahbahrami, A. (2013). A predictive algorithm for multimedia data compression. *Multimedia Systems, 19*(2), 103-115. https://doi.org/10.1007/s00530-012-0282-0

[4] Chen, W. & Shih, C. C. (2012) Architecture of Portable Electronic Medical Records System Integrated with Streaming Media. *Journal of Medical Systems, 36*(1), 25-31. https://doi.org/10.1007/s10916-010-9442-y

[5] Lu, Q., Luo, W., Wang, J., & Chen, B. (2008). Low-complexity and energy efficient image compression scheme for wireless sensor networks. *Computer Networks, 52*(13), 2594-2603. https://doi.org/10.1016/j.comnet.2008.05.006

[6] Lee, S., Lee, T., Jin, G., & Hong, J. (2008). An implementation of wireless medical image transmission system on mobile devices. *Journal of Medical Systems, 32*(6), 471-480. https://doi.org/10.1007/s10916-008-9153-9

[7] Yang, C.-T. & Chu, Y.-Y. (2010). Implementation of a Medical Information Service on Android Mobile Devices. *4th International Conference on New Trends in Information Science and Service Science*, 72-77.

[8] Nimmagadda, Y., Kumar, K., & Lu, Y.-H. (2009). Energy-efficient image compression in mobile devices for wireless transmission. *2009 IEEE International Conference on Multimedia and Expo*, 1278-1281. https://doi.org/10.1109/ICME.2009.5202735

[9] Strutz, T. (2016). Context-Based Predictor Blending for Lossless Color Image Compression. *IEEE Transactions on Circuits and Systems for Video Technology, 26*(4), 687-695. https://doi.org/10.1109/TCSVT.2015.2416611

[10] Lucas, L. F. R., Rodrigues, N. M. M., da Silva Cruz, L. A., & de Faria, S. M. M. (2017). Lossless Compression of Medical Images Using 3-D Predictors. *IEEE Transactions on Medical Imaging, 36*(11), 2250-2260. https://doi.org/10.1109/TMI.2017.2714640

[11] Knezović, J., et al. (2007). Application of Novel Lossless Compression of Medical Images Using Prediction and Contextual Error Modeling. *Collegium antropologicum, 31*(4), 315-319.

[12] NIfTI-1 Data Format-Neuroimaging Informatics Technology Initiative. http://nifti.nimh.nih.gov/nifti-1/

[13] Mustra, M., Delac, K., & Grgic, M. (2008). Overview of the DICOM Standard. *50th International Symposium ELMAR*, 39-44.

[14] Žagar, M., Mlinarić, H., & Knezović, J. (2011). 3D Surface Analysis and Classification in Neuroimaging Segmentation. *Collegium antropologicum, 35*(2), 487-498.

[15] Belina, S., Cuk, V., & Klapan, I. (2009). Virtual endoscopy and 3D volume rendering in the management of frontal sinus fractures. *Collegium antropologicum, 33*(2), 43-51.

[16] Li, X. & Orchard,M. T. (2001) Edge-directed prediction for lossless compression of natural images. *IEEE Transaction Image Processing, 10*(6), 813-817. https://doi.org/10.1109/83.923277

[17] Seemann, T. & Tischer, P. (1997). Generalised locally adaptive DPCM. *Proceedings DCC Data Compression Conference*, 473-488. https://doi.org/10.1109/DCC.1997.582142

[18] Weinberger, M. J., Seroussi, G., & Sapiro, G. (2000). The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS. *IEEE Transactions on Image Processing, 9*(8), 1309-1324. https://doi.org/10.1109/83.855427

[19] Slyz, M. J. & Neuhoff, D. L. (1994). A nonlinear VQ-based predictive lossless image coder. *Proceedings of IEEE Data Compression Conference (DCC'94)*, 304-310. https://doi.org/10.1109/DCC.1994.305938

[20] Medical Image Samples. http://barre.nom.fr/medical/samples/

[21] Wu, X. & Memon, N. (2000). Context-based lossless interband compression--extending CALIC. *IEEE Transactions on Image Processing, 9*(6), 994-1001. https://doi.org/10.1109/83.846242

[22] Santa-Cruz, D. & Ebrahimi, T. (2000). A study of JPEG 2000 still image coding versus other standards. *10th European Signal Processing Conference*, 1-4. https://doi.org/10.1117/12.411564

[23] Klapan, I., Vranjes, Z., Prgomet, D., & Lukinović, J. (2008). Application of advanced virtual reality and 3D computer assisted technologies in tele-3Dcomputer-assisted surgery in rhinology. *Collegium antropologicum, 32*(1), 217-226.

Contact information:

Josip KNEZOVIĆ, associated prof. dr. sc.
University Zagreb,
Faculty of Electrical Engineering and Computing,
Unska 3, 10000 Zagreb, Croatia
josip.knezovic@fer.hr

Martin ŽAGAR, assistant prof. dr. sc.
Rochester Institute of Technology Croatia,
Damira Tomljanovića Gavrana 15, 10000 Zagreb, Croatia
martin.zagar@croatia.rit.edu

Hrvoje MLINARIĆ, associated prof. dr. sc.
(Corresponding author)
University Zagreb,
Faculty of Electrical Engineering and Computing,
Unska 3, 10000 Zagreb, Croatia
hrvoje.mlinaric@fer.hr