Effects of JPEG XR Compression Settings on Iris Recognition Systems *

Kurt Horvath[1], Herbert Stögner[1], and Andreas Uhl[1,2]

[1] School of CEIT, Carinthia University of Applied Sciences, Austria
[2] Department of Computer Sciences, University of Salzburg, Austria

Contact author e-mail: uhl@cosy.sbg.ac.at

Abstract. JPEG XR is considered as a lossy sample data compression scheme in the context of iris recognition techniques. It is shown that apart from low-bitrate scenarios, JPEG XR is competitive to the current standard JPEG2000 while exhibiting significantly lower computational demands.

1 Introduction

With the increasing usage of biometric systems the question arises naturally how to store and handle the acquired sensor data (denoted as sample data subsequently). In this context, the compression of these data may become imperative under certain circumstances due to the large amounts of data involved. Among other possibilities (e.g. like compressed template storage on IC cards and optional storage of (encrypted) reference data in template databases), compression technology is applied to sample data in distributed biometric systems, where the data acquisition stage is often dislocated from the feature extraction and matching stage (this is true for the enrolment phase as well as for authentication). In such environments the sample data have to be transferred via a network link to the respective location, often over wireless channels with low bandwidth and high latency. Therefore, a minimisation of the amount of data to be transferred is highly desirable, which is achieved by compressing the data before transmission and any further processing. As an alternative, the application of feature extraction before transmission looks promising due to the small size of template data but cannot be done under most circumstances due to the prohibitive computational demand of these operations (current sensor devices are typically far too weak to support this while compression can be done e.g. in dedicated low power hardware).

While current international standards define the application of JPEG2000 for lossy iris sample data compression, we focus in this paper on the corresponding application of the recent JPEG XR still image coding standard. We

* This work has been partially supported by the Austrian Science Fund, project no. L554-N15.

experimentally compare the achieved results to a JPEG2000 based (and therefore standard conformant) environment. In particular, we investigate the effects of applying different settings concerning the use of the optional Photo Overlap Transform (POT) as a part of JPEG XR's Lapped Biorthogonal Transform (LBT) with respect to iris recognition accuracy. In Section 2, we review related standards and literature in the area of lossy iris sample data compression. Section 3 presents experiments where we first shortly review the four different iris recognition systems employed in this study. Subsequently, JPEG XR basics and the investigated transform settings are briefly explained. Experimental results comparing JPEG XR and JPEG2000 are shown with respect to PSNR (image quality), execution speed, and iris recognition accuracy in terms of EER. Section 4 concludes the paper.

2 Biometric Iris Sample Compression

During the last decade, several algorithms and standards for compressing image data relevant in biometric systems have evolved. The certainly most relevant one is the ISO/IEC 19794 standard on Biometric Data Interchange Formats, where in its former version (ISO/IEC 19794-6:2005), JPEG and JPEG2000 (and WSQ for fingerprints) were defined as admissible formats for lossy compression, whereas for lossless and nearly lossless compression JPEG-LS as defined in ISO/IEC 14495 was suggested. In the most recently published version (ISO/IEC FDIS 19794-6 as of August 2010), only JPEG2000 is included for lossy compression while the PNG format serves as lossless compressor. These formats have also been recommended for various application scenarios and standardised iris images (IREX records) by the NIST Iris Exchange (IREX `http://iris.nist.gov/irex/`) program.

The ANSI/NIST-ITL 1-2011 standard on "Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information" (2nd draft as of February 2011, former ANSI/NIST-ITL 1-2007) supports both PNG and JPEG2000 for the lossless case and JPEG2000 only for applications tolerating lossy compression.

In literature on compressing iris imagery, rectangular as well as polar iris sample data has been considered. With respect to employed compression technology, we find JPEG [1, 8], JPEG2000 [4, 1, 8], and other general purpose compression techniques [8] being investigated. Superior compression performance of JPEG2000 over JPEG is seen especially for low bitrates (thus confirming the choice of the above-referenced standards), however, for high and medium quality JPEG is found still to be competitive in terms of impacting recognition accuracy. Apart from applying the respective algorithms with their default settings and standard configurations, work has been done to optimise the compression algorithms to the application domain: For JPEG2000, we have proposed to invoke RoI coding for the iris texture area [3] whereas the removal of the image background before compression has also been suggested (i.e. parts of the image not being part of the eye like eye-lids are replaced by constant average gray [1]). For JPEG, we have demonstrated an optimisation of quantisation matrices to achieve better matching accuracy compared to the standard values for rectangular iris image data [7] as well as for polar iris images [6].

The recent JPEG XR standard has not yet been investigated in the context of biometric systems. It might represent an interesting alternative to JPEG2000 due to its simpler structure and less demanding implementations in terms of memory and CPU resources.

3 Experiments on Compressing Iris Sample Data

3.1 Iris Recognition and Iris Database

It is crucial to assess the effects of compressing iris samples using a set of different iris recognition schemes since it can be expected that different feature extraction strategies will react differently when being confronted with compression artefacts and reduced image quality in general.

Many iris recognition methods follow a quite common scheme close to the well known and commercially most successful approach by Daugman. In our pre-processing approach (following e.g. Ma et al. [9]) we assume the texture to be the area between the two almost concentric circles of the pupil and the outer iris. These two circles are found by contrast adjustment, followed by Canny edge detection and Hough transformation. After the circles are detected, unwrapping along polar coordinates is done to obtain a rectangular texture of the iris. In our case, we always re-sample the texture to a size of 512x64 pixels. Subsequently, features are extracted from this iris texture (which has also been termed polar iris image), we consider the following four techniques in this work:

1. A wavelet-based approach proposed by Ma et al. [9] is used to extract a bit-code. The texture is divided into N stripes to obtain N one-dimensional signals, each one averaged from the pixels of M adjacent rows. We used $N = 10$ and $M = 5$ for our 512x64 pixel textures (only the 50 rows close to the pupil are used from the 64 rows, as suggested in [9]). A dyadic wavelet transform is then performed on each of the resulting 10 signals, and two fixed subbands are selected from each transform. This leads to a total of 20 subbands. In each subband we then locate all local minima and maxima above some threshold, and write a bitcode alternating between 0 and 1 at each extreme point. Using 512 bits per signal, the final code is then 512x20 bit. Matching different codes is done by computing the Hamming Distance.
2. Again restricting the texture to the same $N = 10$ stripes as described before, we use a custom C implementation similar to Libor Masek's Matlab implementation[3] of a 1-D version of the Daugman iris recognition algorithm as the second feature extraction technique. A row-wise convolution with a complex Log-Gabor filter is performed on the texture pixels. The phase angle of the resulting complex value for each pixel is discretized into 2 bits. Those 2 bit of phase information are used to generate a binary code, which

[3] `http://www.csse.uwa.edu.au/~pk/studentprojects/libor/sourcecode.html`

therefore is 512x20 bit (again, Hamming Distance can be used for similarity determination).

3. The third algorithm has been proposed by Ko et al. [5]. Here feature extraction is performed by applying cumulative-sum-based change analysis. The algorithm discards parts of the iris texture, from the right side [45^o to 315^o] and the left side [135^o to 225^o], since the top and bottom of the iris are often hidden by eyelashes or eyelids. Subsequently, the resulting texture is divided into basic cell regions (these cell regions are of size 8×3 pixels). For each basic cell region an average gray scale value is calculated. Then basic cell regions are grouped horizontally and vertically. It is recommended that one group should consist of five basic cell regions. Finally, cumulative sums over each group are calculated to generate an iris-code. If cumulative sums are on an upward slope or on a downward slope these are encoded with 1s and 2s, respectively, otherwise 0s are assigned to the code. In order to obtain a binary feature vector (to enable Hamming Distance computation for comparison) we rearrange the resulting iris-code such that the first half contains all upward slopes and the second half contains all downward slopes. With respect to the above settings the final iris-code consists of 2400 bits.
4. Finally, we employ the feature extraction algorithm of Zhu et al. [10] which applies a 2-D wavelet transform to the polar image first. Subsequently, first order statistical measures are computed from the wavelet subbands (i.e. mean and variance) and concatenated into a feature vector. The similarity between two of these real-valued feature vectors is determined by computing the corresponding l^2-Norm.

The following dataset is used in the experiments:

CASIAv3 Interval database[4] consists of NIR images with 320×280 pixels in 8 bit grayscale .jpeg format (high quality) of 249 persons, where for many persons both eyes are available which leads to 391 (image) classes overall.

For intra-class matches (genuine user matches), we consider all possible template pairs for each class (overall 8882 matches), while for inter-class matches (impostor matches) the first two templates of the first person are matched against all templates of the other classes (overall 2601 matches).

3.2 Compression Techniques: JPEG XR and JPEG2000

Originally developed by Microsoft and termed “HD Photo”, JPEG XR got standardized by ITU-T and ISO in 2009 [2], which makes it the most recent still image coding standard. The original scope was to develop a coding scheme targeting “extended range” applications which involves higher bit-depths as currently supported. However, much more than 10 years after JPEG2000 development and 10 years after its standardisation it seems to be reasonable to look for a new

[4] `http://www.cbsr.ia.ac.cn/IrisDatabase.htm/`

coding standard to eventually employ "lessons learnt" in JPEG2000 standardisation. In particular, the focus is on a simpler scheme which should offer only the amount of scalability actually required for most applications (as opposed to JPEG2000 which is a rather complex scheme offering almost unconstraint scalability). JPEG XR shares many properties with JPEG and JPEG2000 but exhibits also elements of the recent H.264 video standardisation [2].

JPEG XR is a transform coding scheme showing the classical three-stage design: transform, quantisation, and entropy encoding. JPEG XR supports lossless to lossy compression of up to 32 bits per colour channel. The transform operates on macroblocks consisting of 16 (arranged in 4 by 4) 4×4 pixel blocks. The first stage of the integer-based transform allowing for perfect reconstruction is applied to all 4×4 pixel blocks of a macroblock. Subsequently, the resulting coefficients are partitioned into 240 "high pass (HP) coefficients" and 16 coefficients corresponding to the lowest frequency in each block. The latter are aggregated into a square data layout (4 x 4 coefficients) onto which the transform is applied for a second time. The result are 15 "low pass (LP) coefficients" and a single "DC" coefficient (per macroblock). It is interesting to note that the concept of recursively applying a filtering operation is "borrowed" from the wavelet transform. Obviously, this also corresponds to three scalability layers: DC, LP, and HP coefficients, similar to the scans being built in the spectral selection JPEG progressive mode.

In fact, the transform used in JPEG XR is more complicated as compared to JPEG, it is a so-called "two-stage lapped biorthogonal transform (LBT)" which is actually composed of two distinct transforms: The Photo Core Transform (PCT) and the Photo Overlap Transform (POT). The PCT is similar to the widely used DCT and exploits spatial correlation within the 4 x 4 pixels block, however, it suffers from the inability to exploit inter-block correlations due to its small support and from blocking artifacts at low bitrates. The POT is designed to exploit correlations across block boundaries as well as mitigate blocking artifacts.

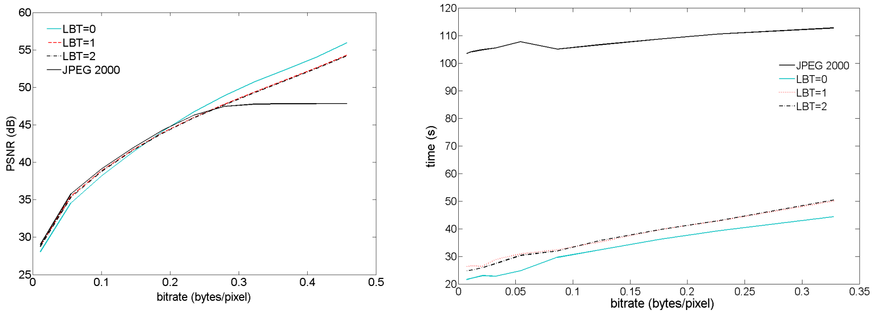
Each stage of the transform can be viewed as a flexible concatenation of POT and PCT since the POT is functionally independent of the PCT and can be switched on or off, as chosen by the encoder (this is signalled by the encoder in the bitstream). There are three options: disabled for both PCT stages, enabled for the first PCT stage but disabled for the second PCT stage, or enabled for both PCT stages.

Since our experiments are focused on the evaluation of those three options concerning POT employment, we do not describe the subsequent JPEG XR stages in the following, please consult the standard or related publications with respect to this issue [2]. For experimentation, we use the official JPEG-XR reference software 1.8 (as of September 2009) and for JPEG2000 compression, imagemagick 8.6.6.0.4-3 (employing libJASPER 1.900.1-7+b1) is used with standard settings.

3.3 Experimental Results

For enabling a fair comparison in the experiments, the same bitrate has to be set in JPEG XR and JPEG2000. While this is straightforward in JPEG2000, JPEG XR suffers from the same weakness as JPEG being unable to explicitly specify a target bitrate. Therefore we have employed a wrapper-program, continuously adapting the JPEG XR quantisation factors (set to identical values for DC, LP, and HP band as used in the default settings) to achieve a certain target bitrate (given in bytes per pixel bpp).

In Fig. 1.a we compare PSNR performance averaged over all images in the considered dataset. Up to 0.2 bpp, JPEG2000 provides the highest values. In this bitrate range, applying no POT (LBT= 0) clearly gives the worst results (PSNR is about 1dB reduced as compared to JPEG2000). Applying POT for the first (LBT= 1) or both transform stages (LBT= 2) leads to almost identical results across the entire bitrate range, up to 0.2bpp PSNR quality is only slightly below that of JPEG2000.



(a) PSNR

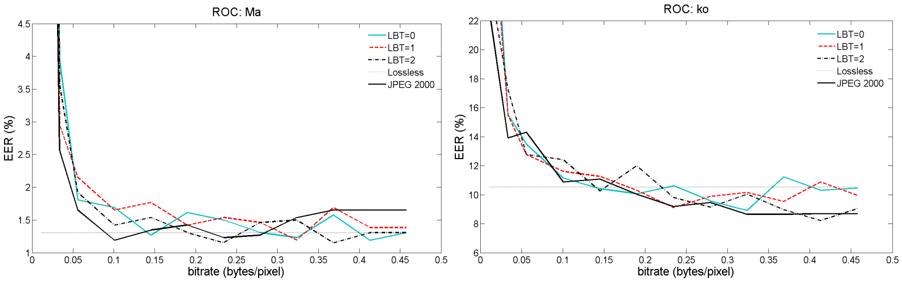
(b) Speed

Fig. 1. Comparing JPEG XR and JPEG2000 in terms of PSNR and Execution Speed.

The situation is different for higher bitrates. JPEG2000 saturates from 0.3bpp upwards due to the employed irreversible 9/7 transform and is clearly outperformed by all JPEG XR settings. Interestingly, for bitrates larger than 0.2bpp, applying no POT gives the best PSNR values, which is explained by the fact that POT application is targeted to optimise data for human perception but not for numerical error minimisation.

Fig. 1.b shows a comparison of execution timings for compressing the entire dataset. We note that depending on the target bitrate considered, JPEG XR is faster by a factor of 2-5 as compared to JPEG2000 (target bitrate optimisation is disabled for this evaluation). This result underlines that JPEG XR could be an interesting alternative to JPEG2000 in biometric environments, especially in cases with limited CPU resources at the compressing site.

In the following, we will investigate the impact compression of one template involved in matching has on the recognition performance of the four iris recognition systems considered (e.g., the sample data acquired by the sensor is compressed and sent to the feature extraction / matching site). For this purpose, we plot equal error rate (EER, on the vertical axis) for applying compression in



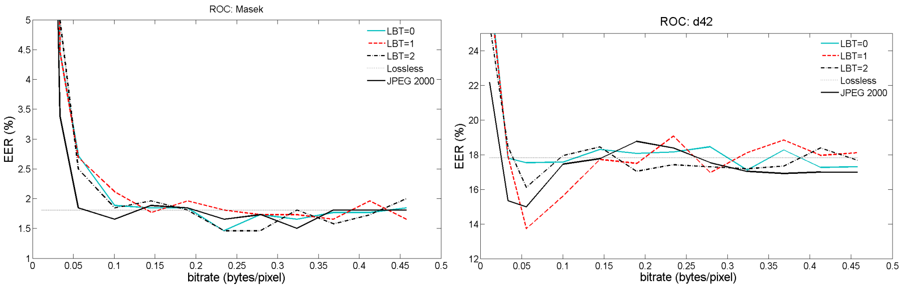
(a) Ma et al.

(b) Ko et al.

Fig. 2. EER for varying bitrates and JEPG XR compression settings.

an entire range of target bitrates (in bpp, on the horizontal axis) and compare JPEG2000 to the three JPEG XR POT employment variants. For reference, also the “Lossless” case (i.e. recognition accuracy in EER without any compression applied) is indicated as a horizontal line in Figs. 2 and 3.

For the algorithms of Ma and Masek, JPEG2000 provides the lowest (i.e. best) EER up to a bitrate of 0.15bpp, while for the other two recognition algorithms, no clear tendency can be observed. In particular, for no algorithm there is a clear indication whether application of POT would be beneficial or not. Further, it is interesting to see that for some algorithms and bitranges, the results involving a compressed template are superior to the uncompressed case (e.g. Ko and Masek for bitrates > 0.2bpp, Zhu for bitrates between 0.04 and 0.15). This can be explained by the fact that compression acts as a denoising filter and has been observed in earlier studies as well [6].



(a) Masek

(b) Zhu et al.

Fig. 3. EER for varying bitrates and JEPG XR compression settings.

What is especially interesting to observe, is that PSNR behaviour as shown in Fig. 1.a does not directly propagate to recognition accuracy. While the better PSNR behaviour of JPEG2000 at low bitrates is at least reflected by the results of two algorithms, we do not find any superiority of JPEG XR for higher bitrates. On the other hand it is interesting to see that except for two recognition algorithms at low bitrates, JPEG XR compressed sample data perform almost equivalent to JPEG2000 compressed one. Given the significantly reduced computational demand as shown in Fig. 1.b, JPEG XR can be considered a promising alternative to JPEG2000 in this application scenario and should be considered in future standardisation efforts in the area.

4 Conclusion

We have found that in the context of biometric systems, JPEG XR can be an interesting alternative to the current standard JPEG2000, especially due to its significantly lower computational demand. A minor decrease in EER as compared to JPEG2000 can be seen only for lower bitrates for two out of four iris recognition systems only. For most iris recognition scenarios, compression with JPEG XR has been identified to be quite competitive to compression with JPEG2000.

References

[1] J. Daugman and C. Downing. Effect of severe image compression on iris recognition performance. *IEEE Transactions on Information Forensics and Security*, 3(1):52–61, 2008.

[2] Frederic Dufaux, Gary J. Sullivan, and Touradj Ebrahimi. The JPEG XR image coding standard. *IEEE Signal Processing Magazine*, 26(6):195–199, November 2009.

[3] J. Hämmerle-Uhl, C. Prähauser, T. Starzacher, and A. Uhl. Improving compressed iris recognition accuracy using JPEG2000 RoI coding. In M. Tistarelli and M.S. Nixon, editors, *Proceedings of the 3rd International Conference on Biometrics 2009 (ICB'09)*, volume 5558 of *LNCS*, pages 1102–1111. Springer Verlag, 2009.

[4] R. W. Ives, R. P. Broussard, L. R. Kennell, and D. L. Soldan. Effects of image compression on iris recognition system performance. *Journal of Electronic Imaging*, 17:011015, doi:10.1117/1.2891313, 2008.

[5] J.-G. Ko, Y.-H. Gil, J.-H. Yoo, and K.-I. Chung. A novel and efficient feature extraction method for iris recognition. *ETRI Journal*, 29(3):399 – 401, 2007.

[6] Mario Konrad, Herbert Stögner, and Andreas Uhl. Custom design of JPEG quantization tables for compressing iris polar images to improve recognition accuracy. In M. Tistarelli and M.S. Nixon, editors, *Proceedings of the 3rd International Conference on Biometrics 2009 (ICB'09)*, volume 5558 of *LNCS*, pages 1091–1101. Springer Verlag, 2009.

[7] G.S. Kostmajer, H. Stögner, and A. Uhl. Custom JPEG quantization for improved iris recognition accuracy. In D. Gritzalis and J. Lopez, editors, *Emerging Challenges for Security, Privacy and Trust. Proceedings of the 24th IFIP International Information Security Conference 2009 (IFIP SEC'09)*, volume 297 of *IFIP AICT*, pages 76–86. Springer Verlag, May 2009.

[8] S. Matschitsch, M. Tschinder, and A. Uhl. Comparison of compression algorithms' impact on iris recognition accuracy. In S.-W. Lee and S.Z. Li, editors, *Proceedings of the 2nd International Conference on Biometrics 2007 (ICB'07)*, volume 4642 of *LNCS*, pages 232–241. Springer Verlag, 2007.

[9] L. Ma; T. Tan; Y. Wang; D. Zhang. Efficient iris recognition by characterizing key local variations. *IEEE Transactions on Image Processing*, 13(6):739–750, June 2004.

[10] Y. Zhu, T. Tan, and Y. Wang. Biometric personal identification based on iris patterns. In *Proceedings of the 15th International Conference on Pattern Recognition (ICPR'00)*, volume 2, pages 2801–2804. IEEE Computer Society, 2000.