Experimental Study on Lossless Compression of Biometric Sample Data

Georg Weinhandel and Herbert Stögner
School of Communication Engineering for IT
Carinthia Tech Institute
Klagenfurt, Austria

Andreas Uhl
Department of Computer Sciences
University of Salzburg, Austria
uhl@cosy.sbg.ac.at

Abstract

The impact of using different lossless compression algorithms on the compression ratios and timings when processing various biometric sample data is investigated. In particular, we relate the application of lossless JPEG, JPEG-LS, lossless JPEG2000 and SPIHT, PNG, GIF, and a few general purpose compression schemes to imagery of the following biometric modalities: fingerprint, iris, retina, face, and hand. Results differing from behaviour found with common or textured imagery are specifically discussed.

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), compression technology may be applied to sample data in two stages of the processing chain in classical biometric recognition for example: In distributed biometric systems, 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. Additionally, optional storage of (encrypted) reference data in template databases also may require the data to be handled in compressed form.

Having found that compression of the raw sensor data can be advantageous or even required in certain applications, we have to identify techniques suited to accomplish this task in an optimal manner. In order to maximise the benefit in terms of data reduction, lossy compression techniques have to be applied. However, the distortions introduced by compression artifacts may interfere with subsequent feature extraction and may degrade the matching results. In particular, FRR or FNMR will increase (since features of the data of legitimate users are extracted less accurately from compressed data) which in turn affects user convenience and general acceptance of the biometric system. In extreme cases, even FAR or FMR might be affected. As an alternative, lossless compression techniques can be applied which avoid any impact on recognition performance but are generally known to deliver much lower compression rates. An additional advantage of lossless compression algorithms is that these are often less demanding in terms of required computations as compared to lossy compression technology.

In this work, we experimentally assess the the application of several lossless compression schemes to sample image data of a variety of biometric modalities. In Section 2, we briefly review related work on biometric sample data compression. Section 3 is the experimental study where we first describe the applied algorithms and biometric data sets. Subsequently, results with respect to achieved compression ratios and timings are discussed. Section 4 concludes this work.

2 Biometric 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 recent ISO/IEC 19794 standard on Biometric Data Interchange Formats, where for lossy compression, JPEG and JPEG2000 (and WSQ for fingerprints) are defined as admissible formats, whereas for lossless and nearly lossless compression JPEG-LS as defined in ISO/IEC 14495 is used.

A significant amount of work exists on using compression schemes in biometric systems. However, the attention is almost exclusively focussed on lossy techniques since in this context the impact of compression to recognition accuracy needs to be investigated. For example, in [6] we have investigated the impact of JPEG, JPEG2000, SPIHT, PRVQ, and fractal image compression on recognition accuracy of selected fingerprint and face recognition systems. Similarly, [3] also relates JPEG, JPEG2000, and (WSQ) compression rates to recognition performance of some fingerprint and face recognition systems. While most work is devoted to lossy fingerprint compression (e.g. [5, 9]), also face [2] and iris [1, 7, 4] image data have been discussed.

One of the few results on applying lossless compression techniques exploits the strong directional features in fingerprint images caused by ridges and valleys. A scanning procedure following dominant ridge direction has shown to improve lossless coding results as compared to JPEG-LS and PNG [12].

In the subsequent experimental study we will apply a set of lossless compression algorithms to image data from the following biometric modalities: fingerprint, iris, retina, face, and hand. Extensive results with respect to achieved compression rate and required compression time are displayed. Specifically, we focus on results differing from behaviour found with common or textured imagery.

3 Experimental Study

3.1 Setting and Methods

3.1.1 Compression Algorithms

We employ the following 6 dedicated lossless image compression algorithms (JPEG2000 – PNG) and 7 general purpose lossless data compression algorithms [10]:

JPEG2000 JasPer reference implementation[1] of JPEG2000 Part 1, a wavelet-based lossy-to-lossless transform coder [11].

Lossless JPEG LibJPEG[2] with default Huffman tables and $PSV = 1$ [14].

SPIHT lossy-to-lossless zerotree-based wavelet transform codec[3] [8].

JPEG-LS reference encoder LOCO[4] using Median edge detection and subsequent predictive and Golumb encoding (in two modes: run and regular modes) [15].

GIF Java standard GIF converter provided by the "ImageIO" class employing LZW encoding.

PNG Irfan View[5] converter using an LZSS encoding variant.

Huffman Coding byte-based Huffman coding[6].

7z,BZ2,GZ 7z uses LZMA as compression procedure which includes an improved LZ77 and range encoder. BZ2 concatenates RLE, Burrows-Wheeler transform and Huffman coding, GZ uses a combination of LZ77 and Huffman encoding. These three algorithms can be employed with the same software[7].

RAR uses LZSS and solid archiving for the compression of multiple files[8].

UHA supports several algorithms out of which ALZ has been used. ALZ is optimised LZ77 with an arithmetic entropy encoder[9].

ZIP uses the DEFLATE algorithm, similar to GZ[10].

3.1.2 Sample Data

For all our experiments we considered images with 8-bit grayscale information per pixel in .pgm format since all software can handle this format (except for SPIHT which requires a RAW format with removed .pgm headers). Database imagery has been converted into this format if not already given so, colour images have been converted to the YUV format using the Y channel as grayscale image. We use the images in their respective original resolutions and in a standardised scaled version for better comparability: approximately squared formats are scaled to 278×278 pixels, other formats to rectangular 320×240 pixels. The scaled versions are meant to highlight the importance of image properties and features regardless of the resolution of the imagery. In order to be able to assess the potential impact of the specific nature of biometric sample data, we also include a database consisting of common images and a texture database for comparison.

Retina DRIVE database[11] consists of 40 images with 565×584 pixels in 24 bit colour.

Fingerprint FVC 2004 database[12] consists of 3600 images in 8 bit grayscale divided into four separated databases, where each database was collected with a different sensor (DB-1: 640×480 pixels with optical sensor; DB-2: 328×364 pixels with optical sensor of different type; DB-3: 300×480 pixels with thermal sweeping sensor; DB-4: 288×384 pixels, synthetic fingerprints).

Iris MMU1 database[13] consists of 450 images with 320×240 pixels in 24 bit grayscale.

Face FERET database[14] consists of 1002 images with 512×768 pixels in 24 bit colour.

Hand Salzburg Computer Science Department handprint database [13] consists of 86 images with 4250×5850 pixels in 8 bit grayscale.

Textures VisTex database[15] consists of 179 images with 512×512 pixels in 24 bit colour; USC-SIP database[16] consists of 37 images with 512×512 pixels in 8 bit grayscale.

[1] http://www.ece.uvic.ca/ mdadams/jasper/
[2] ftp://ftp.cs.cornell.edu/pub/multimed/ljpg.tar.Z
[3] http://www.cipr.rpi.edu/research/SPIHT
[4] http://www.hpl.hp.com/loco
[5] http://www.irfanview.com/
[6] http://bijective.dogma.net/
[7] http://downloads.sourceforge.net/sevenzip/7z457.exe
[8] http://www.rarlab.com/rar/wrar370d.exe
[9] ftp://ftp.sac.sk/pub/sac/pack/uharc06b.zip
[10] ftp://ftp.univ-paris1.fr/pub/pc/zip23xn.zip
[11] http://www.isi.uu.nl/Portal/
[12] http://biometrics.cse.msu.edu/fvc04db/
[13] http://pesona.mmu.edu.my/ ccteo/
[14] http://face.nist.gov/colorferet/
[15] http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html
[16] http://sipi.usc.edu/database/database.cgi?volume=textures

Images ZuBuD database[17] consists of 1002 images with 640×480 pixels in 24 bit colour.

(a) Retina (b) Face (c) Image (d) Texture 1 (e) Texture 2

Figure 1. Example images from the used databases.

Figures 1 and 2 provide one example image from each database.

(a) Iris (b) Hand (c) DB1 (d) DB2 (e) DB3 (f) DB4

Figure 2. Further example images from the used databases.

3.2 Results

3.2.1 Compression Ratio

In the subsequent plots, we display the achieved compression ratio on the y-axis, while giving results for different databases or compression algorithms on the x-axis. When comparing all databases under the compression of a single algorithm, LOCO provides a prototypical result as shown in Fig. 3.

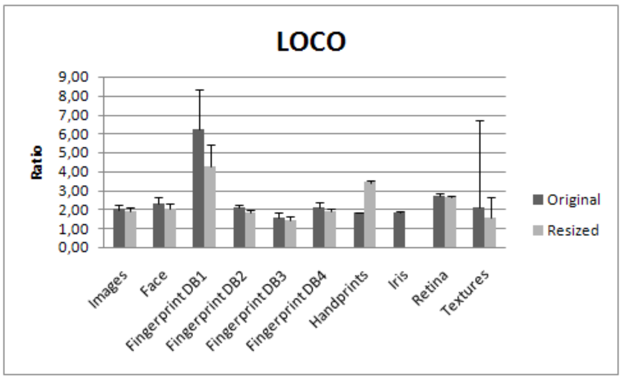


Figure 3. Compression ratios achieved by LOCO.

For most images, we result in a compression ratio of about 2. As it is to be expected, the larger original data sets give higher ratios as compared to the rescaled ones (except for the handprints: here the high resolution originals contain a high amount of noise which is significantly reduced by the rescaling operation resulting in better compression). The fingerprint DB1 images allow a much higher ratio due to the high amount of uniform background present in those images (see Fig. 2.c), due to the varying amount of background we also result in a higher standard deviation as indicated by the error bars. Even higher result variability is observed for the Textures which is due to the very inhomogeneous nature of the dataset caused by the fusion of two databases (see Figs. 1.d and 1.e).

While for most compression algorithms the ranking and relative performance of achieved compression ratios for the different databases is identical or at least similar to LOCO, JPEG2000 behaves differently. Fig. 4 shows the corresponding compression ratios.

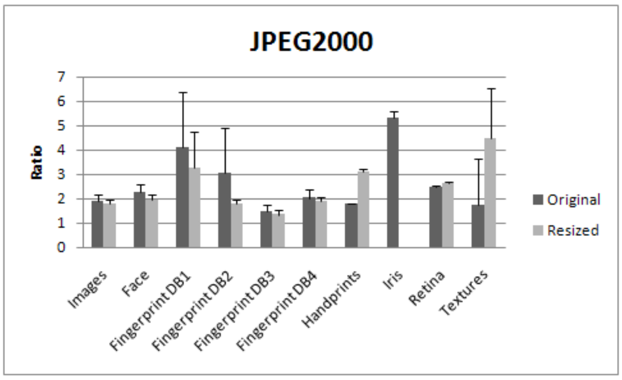


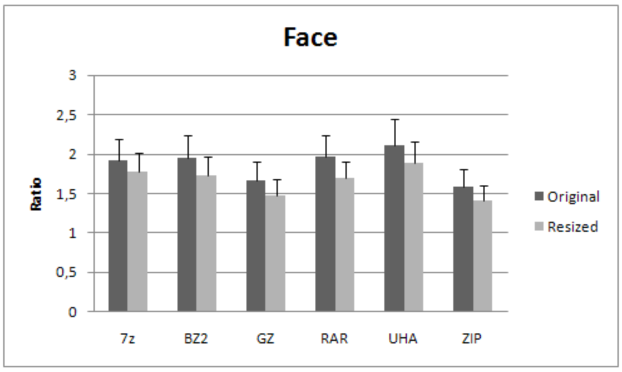
Figure 4. Compression ratios achieved by JPEG2000.

We notice that the results for iris images and fingerprint DB2 images are clearly better than the other results (compared to the LOCO results and those of the other databases except for DB1). This effect is only observed for JPEG2000. Additionally, we observe a higher compression ratio for the resized textures as compared to the ratio for the original ones. This effect is also due to reduced noise in the natural textures and can be observed in similar manner for GIF, PNG, 7z, BZ2, GZ, ZIP, and RAR.

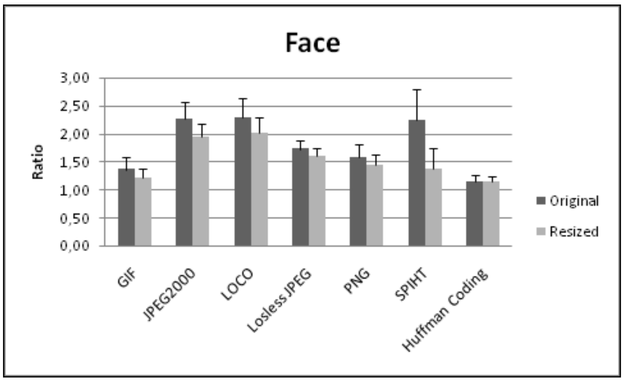
When comparing all compression algorithms when applied to a single database, face images provide prototypical results in this case, also very similar to the common images, since face imagery shares many properties with common image material. Fig. 5 shows corresponding results.

JPEG2000, LOCO, and SPIHT perform almost identical and give the best results with ratio above 2 closely followed by UHA. RAR, B2Z, and 7z are next. The "mid range results" slightly above ratio 1.5 are delivered by lossless JPEG, GZ, ZIP, and PNG. Poorest results below ratio 1.5 are seen for GIF and Huffman coding. While the absolute compression ratios change from database to database, the ranking and relative behaviour of the different algorithms are very similar for many image types (which is especially true for the general purpose compression schemes which deliver

[17] http://www.vision.ee.ethz.ch/showroom/zubud/index.en.html



(a) general purpose compression



(b) image compression

Figure 5. Compression ratios for face images.

robust results fairly independent of image content). In the following, we only discuss significant deviations from the described scenario.

Fig. 6 shows results for compressing fingerprint DB1 images. In this case, LOCO is the only algorithm achieving compression ratio of more than 6. Closely following are UHA, 7z, and BZ2 (not shown) while JPEG2000 is among the weak performing techniques for these images. On the other hand, PNG results are almost on the same level as SPIHT results here.

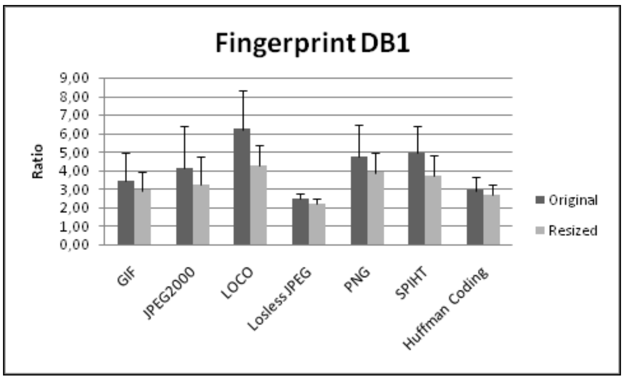
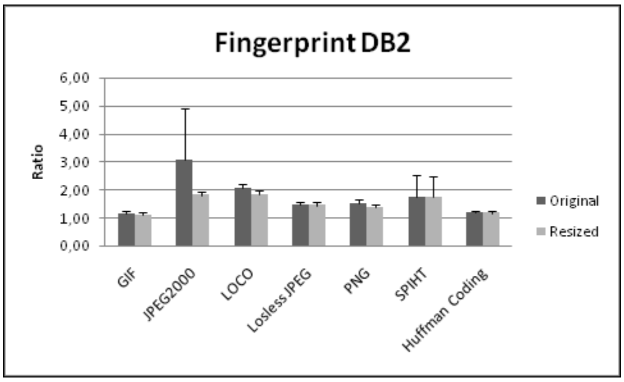
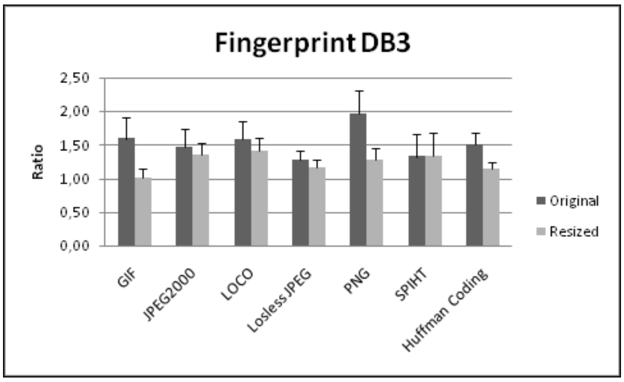


Figure 6. Compression ratios for fingerprint DB1 images.

In Fig. 7, we display results when compressing fingerprint DB2 and DB3 images. Again, we notice different results as compared to the "reference" behaviour for face imagery (see Fig. 5). Fig. 7.a reveals that JPEG2000 significantly outperforms all other algorithms, which do not even reach ratio 2 (except for LOCO). This is also true for the general purpose compression schemes which are not shown.



(a) DB2



(b) DB3

Figure 7. Compression ratios for fingerprint images.

Fig. 7.b shows that the situation is again different for DB3 images. Here, PNG is the best of the image compression algorithms with ratio slightly below 2, however, for this database the general purpose algorithms all except ZIP achieve compression ratio of 2 or even slightly higher (not shown) with UHA performing best.

Finally, iris images exhibit again fairly different results with respect to the top performing algorithm as shown in Fig. 8. JPEG2000 provides compression ratio of more than 5, the next ranked algorithm (LOCO) gives ratio slightly below 2.

In Table 1 we display the best and worst compression algorithm for each image database considered in this study, where all images are considered in their respective original resolution. Table 2 shows analogous results for the scaled image material.

For original resolutions, we notice that the differences between best and worst achieved compression ratio is rather

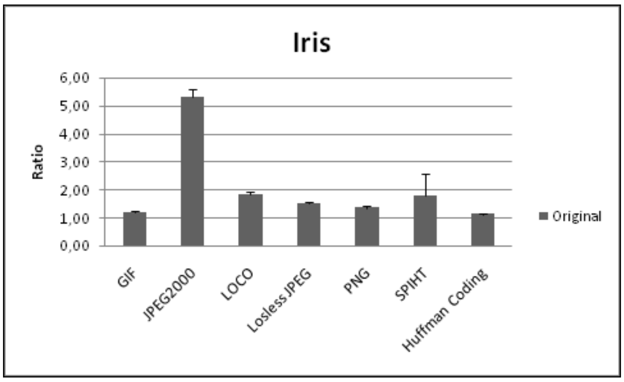


Figure 8. Compression ratios for iris images.

high compared among all databases. Surprisingly GIF is listed several times as the worst algorithm exhibiting compression ratios of slightly above 1.0 only. The highest overall compression ratios are obtained by LOCO, JPEG2000, and UHA for DB1 (ratio 6.20), iris (ratio 5.25), and texture images (ratio 3.98), respectively.

	Best	Ratio	Worst	Ratio
Retina	LOCO	2.73	Huffman	1.46
Iris	JPEG2000	5.25	Huffman	1.06
Face	LOCO	2.27	Huffman	1.20
Hand	SPIHT	1.79	GIF	1.15
DB1	LOCO	6.20	lossless JPEG	2.50
DB2	JPEG2000	3.04	GIF	1.05
DB3	UHA	2.26	lossless JPEG	1.30
DB4	SPIHT	2.17	GIF	1.08
Image	LOCO	1.98	GIF	1.04
Texture	UHA	3.98	GIF	1.00

Table 1. Best and worst compression algorithm for each database (in original resolution) with corresponding achieved compression ratio.

The situation gets slightly different for image material which has been scaled to identical resolution as shown in Table 2. 7z is found to be best performing for two databases now, while SPIHT is no longer seen in the table. Still, the same databases achieve the highest compression ratio (Iris, DB1, and Texture) which indicates that it is not the resolution that makes these images well suited for compression. As already noted, the compression ratios achieved for handprints and textures are better in reduced resolution which is due to the reduced noise due to downscaling.

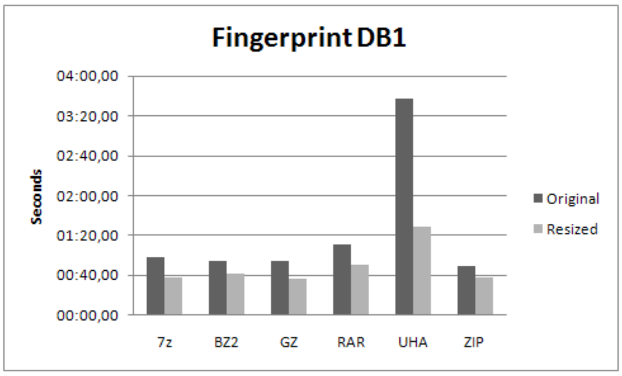
3.2.2 Runtime Performance

Since relative timings turned out to be rather independent of the type of biometric modality (i.e. image type), we present results for only a single database. The timings shown in Fig. 9 are obtained by compressing the entire fingerprint DB1 database.

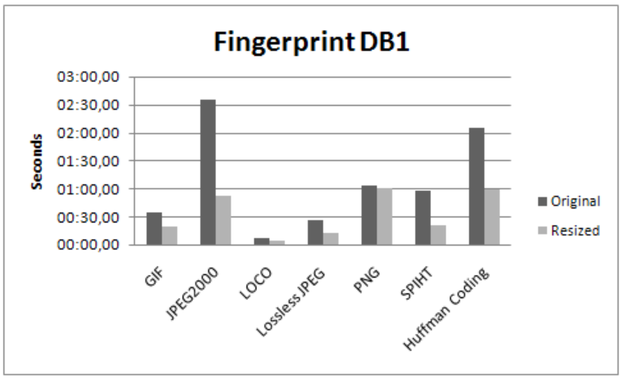
LOCO is by far the fastest software, lossless JPEG is second and GIF third ranked. On the other end of the spectrum,

	Best	Ratio	Worst	Ratio
Retina	LOCO	2.68	Huffman	1.47
Iris	JPEG2000	5.25	Huffman	1.06
Face	LOCO	2.02	Huffman	1.20
Hand	LOCO	3.46	Huffman	1.41
DB1	7z	4.45	lossless JPEG	2.10
DB2	JPEG2000	1.91	GIF	1.04
DB3	7z	1.47	GIF	1.01
DB4	JPEG2000	1.91	GIF	1.04
Image	LOCO	1.87	GIF	0.99
Texture	JPEG2000	4.48	Huffman	1.06

Table 2. Best and worst compression algorithm for each database (in scaled resolution) with corresponding achieved compression ratio.



(a) general purpose compression



(b) image compression

Figure 9. Timings.

UHA and JPEG2000 are significantly slower as the other algorithms, followed by Huffman coding. This applies to the higher resolution originals. For the rescaled versions, the overall trend is similar, but some algorithms benefit more of the lower resolution than others (e.g. JPEG2000, UHA, SPIHT, Huffman coding).

It has to be noted that the timings shown of course do depend of the actual software being employed for compression which can significantly vary in terms of being optimised. However, apart from the results for Huffman coding and JPEG20000 (which seem to be unexpectedly slow, e.g. JPEG2000 is expected to behave similar to SPIHT in term of runtime performance) the results correspond to a certain extent to the algorithmic complexity and can serve as a rough guideline for practical usage.

4 Conclusion and Future Work

Overall, LOCO seems to be the best compromise between compression results (where results are among the top three in almost all cases) and runtime performance (where LOCO turns out to be the fastest software). Therefore, the employment of JPEG-LS in biometric systems can be recommended for most scenarios which confirms the standardisation done in ISO/IEC 19794. For image data sets where other algorithms give significantly better compression results the higher computational demand of other schemes might be justified, e.g. for JPEG2000 applied to iris images and fingerprints DB2 and eventually for PNG applied to fingerprints DB3.

Acknowledgements

Most of the work described in this paper has been done in the scope of the ILV "Compression Technologies and Data Formats" (winter term 2008/2009) in the master program on "Communication Engineering for IT" at Carinthia Tech Institute. The artificial name "Georg Weinhandel" represents the following group of students working on this project: Brezjak Mario, Dochie Oana, Fabach Martin, Garz Christian, Hinteregger Thomas, Huber Reinhard, Kuttruff Marc, Lesjak Daniel, Mair Hannes, Mrak Matic, Pichler Alexander, Printschler Martin. This work has been partially supported by the Austrian Science Fund, project no. L554-N15.

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] K. Delac, M. Grgic, and S. Grgic. Effects of JPEG and JPEG2000 compression on face recognition. In *Proceedings of ICAPR 2005*, volume 3687 of *LNCS*, pages 136–145. Springer-Verlag, 2005.

[3] W. Funk, M. Arnold, C. Busch, and A. Munde. Evaluation of image compression algorithms for fingerprint and face recognition systems. In J. Cole and S. Wolthusen, editors, *Proceedings from the Sixth Annual IEEE Systems, Man and Cybernetics (SMC) Information Assurance Workshop*, pages 72–78. IEEE Computer Society, June 2006.

[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] R.C. Kidd. Comparison of wavelet scalar quantization and JPEG for fingerprint image compression. *Journal of Electronic Imaging*, 4(1):31–39, 1995.

[6] A. Mascher-Kampfer, H. Stögner, and A. Uhl. Comparison of compression algorithms' impact on fingerprint and face recognition accuracy. In C.W. Chen, D. Schonfeld, and J. Luo, editors, *Visual Communications and Image Processing 2007 (VCIP'07)*, number 6508 in Proceedings of SPIE, pages 650810–1 – 65050N–10, San Jose, CA, USA, January 2007. SPIE.

[7] 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.

[8] Amir Said and William A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3):243–249, June 1996.

[9] Barry G. Sherlock and Donald M. Monro. Optimized wavelets for fingerprint compression. In *Proceedings of the 1996 International Conference on Acoustics, Speech and Signal Processing (ICASSP'96)*, Atlanta, GA, USA, May 1996.

[10] J.A. Storer. *Image and Text Compression*. The Kluwer international series in engineering and computer science. Kluwer Academic Publishers Group, Boston, 1992.

[11] D. Taubman and M.W. Marcellin. *JPEG2000 — Image Compression Fundamentals, Standards and Practice*. Kluwer Academic Publishers, 2002.

[12] J. Thärna, K. Nilsson, and J. Bigun. Orientation scanning to improve lossless compression of fingerprint images. In J. Kittler and M.S. Nixon, editors, *Proceedings of AVBPA*, volume 2688 of *LNCS*, pages 343–350. Springer Verlag, 2003.

[13] Andreas Uhl and Peter Wild. Personal recognition using single-sensor multimodal hand biometrics. In A. Elmoataz, O. Lezoray, F. Nouboud, and D. Mammass, editors, *Image and Signal Processing. Proceedings of ICISP 2008*, volume 5099 of *LNCS*, pages 396–404. Springer Verlag, 2008.

[14] G.K. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):30–44, 1991.

[15] M. Weinberger, G. Seroussi, and G. Sapiro. Lossless image compression algorithm: Principles and standardization into JPEG-LS. *IEEE Transactions on Image Processing*, 9(8):1309–1324, August 2000.