BENCHMARKING BINARISATION SCHEMES FOR DEEP FACE TEMPLATES

P. Drozdowski[*†], *F. Struck*[*], *C. Rathgeb*[*] *and C. Busch*[*]

[*] da/sec - Biometrics and Internet Security Research Group, Hochschule Darmstadt, Germany
[†] Norwegian Biometrics Laboratory, NTNU, Gjøvik, Norway

{pawel.drozdowski,christian.rathgeb,christoph.busch}@h-da.de
florian.struck@stud.h-da.de

ABSTRACT

Feature vectors extracted from biometric characteristics are often represented using floating point values. It is, however, more appealing to store and compare feature vectors in a binary representation, since it generally requires less storage and facilitates efficient comparators which utilise intrinsic bit operations. Furthermore, the binary representations are very often necessary for some specific application scenarios, e.g. template protection and indexing.

In recent years, usage of deep neural networks for facial recognition has vastly improved the biometric performance of said systems. In this paper, various binarisation schemes are applied to such feature vectors and benchmarked for biometric performance. It is shown that with only a negligible drop in biometric performance, the storage space and computational requirements can be vastly decreased.

Index Terms— Biometrics, face recognition, binarisation, deep face templates

1. INTRODUCTION

Face is one of the most widely used biometric characteristics. Various methods have been proposed over the span of last three decades [1, 2]. In recent years, methods based on deep learning (*e.g.* [3, 4, 5, 6]) have been proposed, and significantly improved on the biometric performance of the heretofore existing methods. With this improved biometric performance, face has become an attractive characteristic for large-scale identification systems.

The deep face feature representations typically involve float-valued vectors, for which the template comparison is performed using metrics such as Euclidean distance (L^2 norm) or Chi-square distance (χ^2). Those metrics are computationally expensive – thus creating a potential efficiency bottleneck for large-scale biometric identification systems, where 1:N template comparisons are performed during lookup. Additionally, transmission of such feature vectors from low-cost mobile devices to central systems requires a compact encoding, specifically when the bandwith of mobile networks is limited. *Binarisation* of feature vectors offers an attractive alternative – such templates can be stored efficiently and be compared quickly in the Hamming domain utilising intrinsic CPU operations (*i.e.* xor and $\mathrm{popcount}$) [7].

Over time, many methods of binarising data have been proposed, mostly with template protection as motivation (to transform the features to certain input forms required by the different cryptographic primitives) [8] and shown to work with, among others, classical facial recognition systems. However, it is unknown, whether or not those approaches are suitable for the vectors produced by deep learning based face feature extractors and their potential biometric performance degradation due to information loss has not been studied thoroughly. With this uncertainty as the motivation, the main contribution of this paper is such a benchmark, where various binarisation methods are evaluated in terms of biometric performance and computational workload incurred at the comparison stage.

The remainder of this paper is organised as follows: Section 2 introduces the related work. In section 3, binarisation schemes for deep facial templates are described. In section 4, the experimental setup and results are presented, while section 5 contains a summary of the paper and future work items.

2. RELATED WORK

In the recent decade, several data binarisation approaches have been proposed. Kevenaar *et al.* [9] extract the most reliable components of facial feature vectors and binarise them for use in a template protection scheme. Chen *et al.* [10] present a detection rate optimized bit allocation (DROBA) principle, which is biometric characteristic-agnostic. Based on the discriminative power of the features, it assigns more or fewer bits to them during binarisation, thus improving the biometric performance of the binarised feature representation. Bringer *et al.* [11] transform fingerprint minutiae set using a vicinity-based approach, which in addition to producing a compact feature representation, also exhibits self-alignment property. When presenting a novel fingerprint minutiae representation scheme, Cappelli *et al.* [12] note that it can also

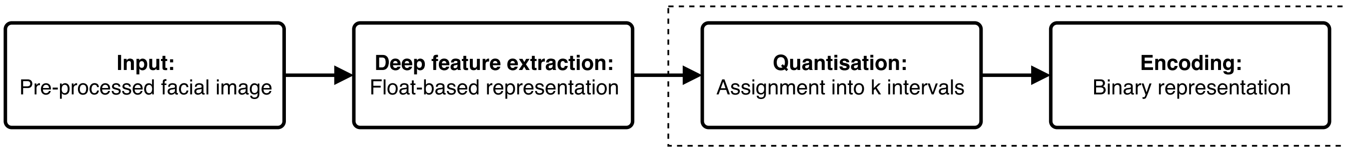


Fig. 1: Processing chain

operate in binarised mode, without significantly decreasing the biometric performance of the scheme. Lee *et al*. [13] binarise facial PCA/ERE-based templates using a generalised Linnartz and Tuyl's quantisation index modulation (QIM) scheme for the purpose of template protection. Chen *et al*. [14] present a generic (for arbitrary characteristics with float-valued feature vectors) binarisation scheme using pairwise adaptive phase quantization and long-short pairing strategy. Lim *et al*. [15] describe a DROBA-based approach, in which bit statistics (reliability and discriminability) to improve the biometric performance of the binarised representation of facial features. In Lim *et al*. [16], the authors propose two new encoding schemes (LSSC and PLSSC – (Partially) Linearly Separable Subcode) which exhibit full-ideal and near-ideal separability capabilities, respectively. Schlett *et al*. [17] describe a simple, yet effective, scheme for binarising multi-scale LBP histograms.

In general, the results presented in the summarised state-of-the-art show, that various float-value based feature representations can be effectively transformed into compact binary strings, without a significant drop in biometric performance, when benchmarked against the original data representation.

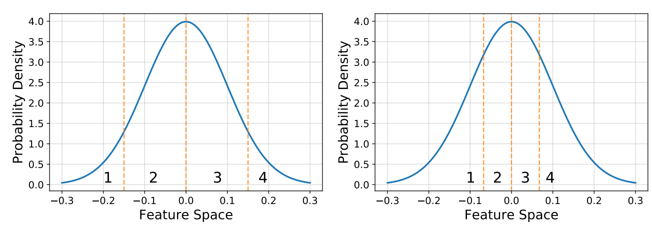
3. BINARISATION OF DEEP FACE TEMPLATES

Figure 1 shows a high-level view of the facial image processing chain used in this paper with the key steps (for this paper) highlighted. First, common pre-processing steps including region of interest detection, alignment and normalisation are applied. The current state-of-the-art deep facial recognition frameworks then extract feature vectors consisting of a pre-defined number of floating point values. Specific details regarding the pre-processing and feature extraction steps are given in subsection 4.1. To avoid computational overhead during comparison stage (computing Euclidean distance with floating-point numbers, as mentioned in section 1), the feature vector can be binarised. Normally, this process consists of two steps [8]: 1. Quantisation (subsection 3.1) and 2. Encoding (subsection 3.2).

3.1. Quantisation

During quantisation, the values from the feature vector are mapped to a number of integer-labelled intervals over the feature space probability density (feature extraction algorithm dependent, obtained via a training set). In this paper, two quantisation schemes listed below are utilised and visualised in figure 2.

- Equal-width quantile: the feature space is divided into segments of equal size (figure 2a)
- Equal-probable quantile: the feature space is divided into segments containing equal population probability mass (figure 2b)



(a) Equal-width (b) Equal-probable

Fig. 2: Quantisation

3.2. Encoding

After quantisation, in the encoding step, the aforementioned quantised intervals (represented as integers) are mapped to short binary strings, which are subsequently concatenated to produce the final feature representation. The dissimilarity of two such templates can be then computed using Hamming distance. The encodings used in this paper are listed below.

- **Boolean** The simplest scheme, where the feature space is quantised into 2 sub-spaces (*i.e*. the resulting encoding is a single 0 or 1).
- **DBR** (Direct Binary Representation) In which the decimal numbers from quantisation are converted directly into their binary representations.
- **BRGC** (Binary Reflected Gray Code [18]) In which the encoding is done so that the distance in the Hamming domain between codewords resulting from successive decimal values is always 1.
- **LSSC** (Linearly Separable Subcode [16]) A more recent approach, which offers ideal separability, *i.e*. the distances between two values are the same in the decimal and Hamming domains.

- **Sparse** A simple scheme, in which the number of encoded bits is equal to that of quantised sub-spaces (k) and only one bit is set to 1 – that corresponding to the sub-space index resulting from the quantisation step. When quantising into larger number of sub-spaces, this can result in a sparse binary feature vector.

Table 1 shows an example with 4 quantisation intervals and the encoding methods described above. Intuitively, there exists a trade-off between the ability to obtain better separation, representation sparsity and the required length of the encoding. In the next section, the schemes are put to test by assessing their biometric performance with deep facial feature vectors.

Table 1: Encoding schemes

Quantisation Interval	**Encoding**				
	Boolean	DBR	BRGC	LSSC	Sparse
1	0	00	00	000	0001
2	1	01	01	001	0010
3	—	10	11	011	0100
4	—	11	10	111	1000

4. EXPERIMENTS

This section contains the evaluation of the binarisation schemes described earlier. In subsection 4.1, the used datasets and experimental setup details are outlined, while the results are presented and discussed in subsection 4.2.

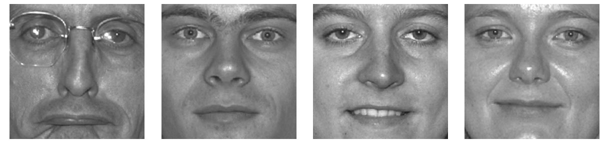
4.1. Experimental Setup

Three commonly used facial datasets, summarised in table 2, were chosen for experiments in this paper. From the FERET dataset, only frontal images were used, while from the AR Face dataset, frontal images without intentional obfuscations (such as sunglasses or scarves) were used. From the FRGC dataset, the complete "Fall2003" subset was used.

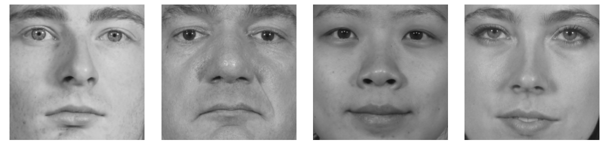
Table 2: Overview of the data used for experiments

Dataset	**Subjects**	**Images**	**Comparisons**	
			Genuine	Impostor
AR Face [19]	133	741	1757	8777
FERET [20]	994	2722	3649	493520
FRGC [21]	370	11358	219851	68264

In the pre-processing stage, the face of a subject is detected and normalised according to eye coordinates detected by the *dlib* landmark detector [22]. Subsequently, the normalised region is cropped to 320×320 pixels and converted to grayscale. Example images from the used datasets (after pre-processing) are shown in figure 3. Thereafter, two state-of-the-art, open-source deep facial recognition frameworks



(a) FERET



(b) FRGC

(c) AR Face

Fig. 3: Example images after pre-processing

(OpenFace [23] and FaceNet [4]) were used to extract features from the images. The resulting representation is a 1-dimensional feature vector containing 128 float values. The frameworks utilised pre-trained (on datasets disjoint from the ones used for the binarisation experiments in this paper) models, made available by their authors, were used.

Suitable thresholds for quantisation intervals are determined via training on the feature space of the AR Face dataset and then used directly in tests on the remaining two datasets. For each binarisation method, all possible template comparisons (verification transactions) were performed to compute the biometric performance of the system. The baseline biometric performance was computed using the aforementioned original, float-based templates, which are compared using squared Euclidean distance.

The metrics used for evaluation were:

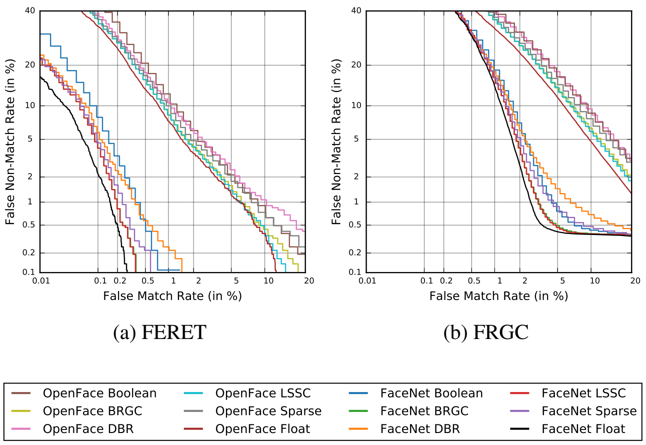
- Biometric performance: Detection error trade-off curve (DET) and equal error rate (EER)

- Template size: bits

- Computational workload: CPU instructions required to perform a single template comparison

4.2. Results

Figure 4 shows DET curves for the performed experiments on the FERET (figure 4a) and FRGC (figure 4b) datasets. It can be seen, that using the FaceNet feature extractor yields results vastly superior to that of OpenFace. Furthermore, it can be seen, that the float-based representation performs only marginally better than the best binarisation schemes.



(a) FERET

(b) FRGC

(c) Legend

Fig. 4: DET curves

In terms of EER, the baseline performance on the FERET dataset is 2.68% and 0.23% EER for OpenFace and FaceNet, respectively, while on the FRGC dataset, it is 7.35% and 2.13% EER for OpenFace and FaceNet, respectively. The experimental results for binarisation schemes are shown in table 3 with best result for each feature extractor/dataset pair marked in bold. Generally, the LSSC encoding has the best performance, which was to be expected, since it offers better separability than the remaining encodings. In most cases, the equal-width quantisation was better than the equal-probable quantisation. In summary, the best quantisation/encoding method pairs suffer only a negligible loss of biometric performance (in terms of EER) against the float-based baseline system: FaceNet loses 0.06 and 0.14 percentage points, while OpenFace loses 0.19 and 0.53 percentage points on FERET and FRGC datasets, respectively.

Table 3: Results (best one(s) for each dataset/extractor pair marked in bold)

Encoding	Quantisation	Size (bits)	Performance (EER)			
			FERET		FRGC	
			FaceNet	OpenFace	FaceNet	OpenFace
Boolean	eq. width	128	0.47%	3.34%	2.85%	9.10%
	eq. probable		0.49%	3.56%	2.80%	9.49%
DBR	eq. width	256	0.52%	3.46%	2.99%	9.31%
		384	0.98%	3.85%	3.72%	10.10%
	eq. probable	256	0.71%	3.65%	3.31%	9.30%
		384	0.76%	3.75%	3.60%	9.36%
BRGC	eq. width	256	**0.29%**	2.87%	**2.32%**	7.95%
		384	0.31%	2.99%	2.36%	8.17%
	eq. probable	256	0.35%	3.20%	2.61%	8.33%
		384	0.36%	3.30%	2.73%	8.42%
LSSC	eq. width	384	**0.29%**	**2.82%**	**2.32%**	**7.88%**
	eq. probable		**0.29%**	2.92%	2.40%	7.97%
Sparse	eq. width	512	0.34%	3.13%	2.54%	8.49%
	eq. probable		0.47%	3.36%	2.92%	8.63%

The binary templates are compared using Hamming distance, *i.e.* by using a binary xor followed by a $\mathrm{popcount}$, both of which are intrinsic operations on the vast majority of modern processors. By storing the binary vectors in arrays of unsigned integers, 64 bits at a time can be handled and then summed up using add operations. Hence, for comparing a binary vector of length $64 * n$, the required number of operations is: $3 * n - 1$. The float-based templates are in this case compared using squared Euclidean distance, which in one dimension corresponds to computing the dot product of the difference between the two feature vectors, *i.e.* a sum of element-wise multiplication between two copies of the difference vector. Table 4 summarises the required instruction numbers for the template representation types and sizes used in this paper's experiments. The original representation requires an order of magnitude more instructions; furthermore, those require floating point arithmetic instead of binary/integer arithmetic. It is therefore clear, that the binarised representation is vastly more efficient computationally.

Table 4: CPU instructions per template comparison

Representation	Instruction Type	Count
128 floats	float sub, mul and add	383
128 bits	binary xor and popcount	5
256 bits		11
384 bits		17
512 bits		23

5. SUMMARY

In this paper, several methods for quantisation and encoding of float-valued deep (OpenFace and FaceNet) feature representation of facial images were benchmarked. In tests on commonly used large facial datasets, the binarised templates suffer only a negligible biometric performance loss against the original, float-valued representation of the deep facial templates, while vastly reducing the template size in bits. As a consequence of the more compact feature vector, and also by being able to use intrinsic CPU operations for template comparison, the computational and storage requirements are vastly reduced. This benchmark thus reveals that the existing binarisation methods can be readily applied to feature vectors produced by deep neural networks.

A promising future work avenue is using the binarised deep face templates to perform (multi-)biometric indexing for further workload reduction in large-scale biometric identification systems.

6. ACKNOWLEDGEMENTS

This work was partially supported by the German Federal Ministry of Education and Research (BMBF), by the Hessen State Ministry for Higher Education, Research and the Arts (HMWK) within Center for Research in Security and Privacy (CRISP), and the LOEWE-3 BioBiDa Project (594/18-17).

7. REFERENCES

[1] A. K. Jain and S. Z. Li, *Handbook of face recognition*, Springer, 2011.

[2] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, December 2003.

[3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *2014 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2014, pp. 1701–1708.

[4] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 815–823.

[5] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2015, pp. 2892–2900.

[6] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Conf.*, 2015, vol. 1.

[7] C. Rathgeb, N. Buchmann, H. Hofbauer, H. Baier, A. Uhl, and C. Busch, "Methods for accuracy-preserving acceleration of large-scale comparisons in CPU-based iris recognition systems," *IET Biometrics*, pp. 1–18, 2017.

[8] M.-H Lim, A. B. J. Teoh, and J. Kim, "Biometric feature-type transformation: Making templates compatible for secret protection," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 77–87, September 2015.

[9] T. A. M. Kevenaar, G. J. Schrijen, M. van der Veen, A. H. M. Akkermans, and F. Zuo, "Face recognition with renewable and privacy preserving binary templates," in *IEEE Workshop on Automatic Identification Advanced Technologies (AutoID)*, October 2005, pp. 21–26.

[10] C. Chen, R. N. J. Veldhuis, and T. A. M. Kevenaar A. H. M. Akkermans, "Biometric quantization through detection rate optimized bit allocation," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, no. 1, pp. 1–16, May 2009.

[11] J. Bringer and V. Despiegel, "Binary feature vector fingerprint representation from minutiae vicinities," in *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, September 2010, pp. 1–6.

[12] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia cylinder-code: A new representation and matching technique for fingerprint recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2128–2141, December 2010.

[13] H. Lee, A. B. J. Teoh, H. G. Jung, and J. Kim, "A secure biometric discretization scheme for face template protection," *Future Generation Computer Systems*, vol. 28, no. 1, pp. 218–231, January 2012.

[14] C. Chen and R. Veldhuis, "Binary biometric representation through pairwise adaptive phase quantization," *EURASIP Journal on Information Security*, vol. 2011, no. 1, pp. 1–16, February 2011.

[15] M.-H. Lim, A. Beng J. Teoh, and K.-A. Toh, "An efficient dynamic reliability-dependent bit allocation for biometric discretization," *Pattern Recognition*, vol. 45, no. 5, pp. 1960–1971, 2012.

[16] M.-H. Lim and A. B. J. Teoh, "A novel encoding scheme for effective biometric discretization: Linearly separable subcode," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 300–313, February 2013.

[17] T. Schlett, C. Rathgeb, and C. Busch, "A binarization scheme for face recognition based on multi-scale block local binary patterns," in *Intl. Conf. of the Biometrics Special Interest Group (BIOSIG)*, September 2016, pp. 1–4.

[18] F. Gray, "Pulse code communications," 1953, U.S. Patent 2,632,058.

[19] A. Martínez and R. Benavente, "The AR face database," Tech. Rep. 24, CVC, June 1998.

[20] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Trans. on pattern analysis and machine intelligence*, vol. 22, no. 10, pp. 1090–1104, 2000.

[21] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, et al., "Overview of the face recognition grand challenge," in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2005, vol. 1, pp. 947–954.

[22] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, 2009.

[23] B. Amos, B. Ludwiczuk, and M. Satyanarayanan, "OpenFace: A general-purpose face recognition library with mobile applications," Tech. Rep. CMU-CS-16-118, CMU School of Computer Science, 2016.