Local Features for Forensic Signature Verification

Muhammad Imran Malik[1], Marcus Liwicki[1,2], and Andreas Dengel[1]

[1] German Research Center for Artificial Intelligence
(DFKI GmbH) Kaiserslautern, Germany
firstname.lastname@dfki.de
[2] University of Fribourg, Switzerland
marcus.liwicki@unifr.ch

Abstract. In this paper we present a novel comparison among three local features based offline systems for forensic signature verification. Forensic signature verification involves various signing behaviors, e.g., disguised signatures, which are generally not considered by Pattern Recognition (PR) researchers. The first system is based on nine local features with Gaussian Mixture Models (GMMs) classification. The second system utilizes a combination of scale-invariant Speeded Up Robust Features (SURF) and Fast Retina Keypoints (FREAK). The third system is based on a combination of Features from Accelerated Segment Test (FAST) and FREAK. All of these systems are evaluated on the dataset of the 4NSigComp2010 signature verification competition which is the first publicly available dataset containing disguised signatures. Results indicate that our local features based systems outperform all the participants of the said competition both in terms of time and equal error rate.

Keywords: Signature verification, disguised signatures, forensic handwriting examination, local features, GMM, SURF, FAST, FREAK.

1 Introduction

Signature verification is in focus of research since decades. Traditionally, automatic signature verification is divided into two broad categories, online and offline, depending on the mode of the handwritten input. If both the spatial as well as temporal information regarding signatures are available to the systems, verification is performed on online data. In the case where temporal information is not available and the systems must utilize the spatial information gleaned through scanned or camera captured documents, verification is performed on offline data [1–3].

In many recent works signature verification has been considered as a two-class pattern classification problem [3]. Here an automated system has to decide whether or not a given signature belongs to a referenced authentic author. If the system could not find enough evidence of a forgery from the questioned

A. Petrosino, L. Maddalena, P. Pala (Eds.): ICIAP 2013 Workshops, LNCS 8158, pp. 103–111, 2013.

signature feature vector, it simply considers the signature as genuine belonging to the referenced authentic author, otherwise it declares the signature as forged.

Apart from the above mentioned two class classification paradigm, another important genre of signatures especially for Forensic Handwriting Examiners (FHEs) is the disguised signatures. A disguised signature is written originally by an authentic author but with the purpose of later denial. Here an authentic author disguises his/her signatures to make them look like a forgery. The purpose of disguising signatures can be hundreds, e.g., a person trying to withdraw money from his/her own bank account via offline signatures on the bank check and trying to deny the check signatures after some time, or even making a false copy of his/her will etc., but they appear often in forensic casework. The category of disguised signatures has been addressed during the ICFHR 4NsigComp 2010 [4]. This was the first attempt to include disguised signatures into a signature verification competition. The systems had to decide whether the author wrote a signature in a natural way, with an intension of a disguise, or whether it has been forged by another writer.

In this paper we investigate three local features based methods on the publicly available 4NSigComp2010 signature verification competition data set. The first system is based on nine local features with Gaussian Mixture Models (GMMs) classification. The second system utilizes a combination of scale-invariant Speeded Up Robust Features (SURF) and Fast Retina Keypoints (FREAK). The third system is based on a combination of Features from Accelerated Segment Test (FAST) and FREAK.

The rest of this paper is organized as follows. Section 2 covers some of the important related work. Section 3 explains the three offline signature verification systems considered in this study. Section 4 explains the dataset used in this study, reports on the experimental results and provides a comparative analysis of the results. Section 5 concludes the paper and gives some ideas for future work.

2 Related Work

Signature verification has remained an active field since the last few decades. The state-of-the-art of signature verification from late 1980's to 2000 are presented in [1], and [2]. Later methods have been summarized in [3]. Throughout these years, various classification methods based on global and/or local features have been presented. A majority of these methods have been tested for detection of genuine and forged signatures but disguised signatures are generally neglected apart from some initial research, like [1], and in some comparative studies of local and global feature based methods, like [5].

Note that, unlike disguised signatures, disguised handwriting in general is previously considered in some PR-research like [6]. However, [6] only focuses the classification of disguise versus genuine handwriting which does not completely suffice the needs of FHEs.

Some local features based methods such as Scale Invariant Feature Transform (SIFT) are used for writer retrieval and identification, as in [7], but identification

and retrieval differ from signature verification in their essence, also disguised signatures are not explicitly focused as yet which is a novel aspect of our study. Similarly some improvements in the basic SIFT descriptors have been suggested in recent past for character recognition, such as in [8], and [9]. The SURF Keypoint detector, which we have used to initially identify the signatures' local regions of interest in one of our experiments, in conjunction with SURF keypoint descriptor has been previously used heavily for object and character recognition, such as in [10–12]. Similarly, the FAST keypoint detector, which we have also used to initially identify the signatures' local regions of interest in one of our systems, has been previously used mainly for problems like multiple object tracking [13], object recognition for smart phone platforms [14], and recognition of degraded handwritten characters [15].

The novelty of our work, however, is the way local features are averaged and applied for signature verification to cater the complete verification paradigm including disguised signatures as well as the comparison we have performed among the three approaches with respect to time and memory.

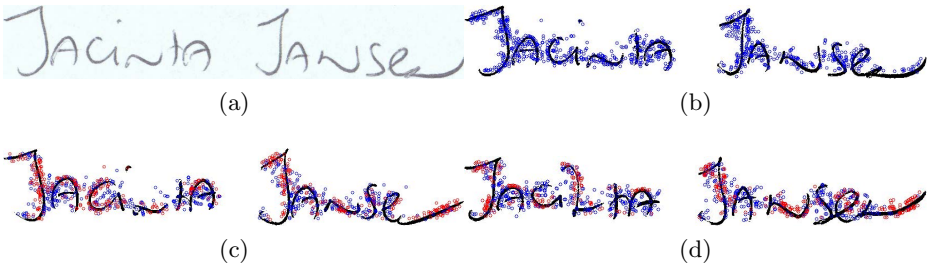

Fig. 1. Some example signatures. (a): A genuine reference signature, (b): Keypoints extracted from the genuine reference signature, (c): Keypoints extracted from a questioned forged signature, (d): Keypoints extracted from a questioned disguised signature (red=matching, blue=non-matching with the questioned signatures keypoints).

3 Local Features Based Systems

In this section we provide a short description of the three local features based offline signature verification systems considered in this study. Note that for the system 1 (i.e., SURF-FREAK) and system 2 (i.e., FAST-FREAK) we followed the same methodology of local interest point/areas detection and description. The difference between the two systems, however, is that in the first system we used the Speeded Up Robust Features (SURF) for detecting the signatures' local areas of interest and in the second system we used the Features from Accelerated Segment Test (FAST) for local interest areas detection. Later, in both the systems, we used the Fast Retina Keypoints (FREAK) descriptors of these local areas of interest in order to perform classification. In the following Section 3.1 we will describe the methodology we followed for the first two systems. Later in

Section 3.2 we will describe the third local features based system we applied in this study.

3.1 Methodology: System 1 and System 2

The proposed approach for signature verification is based on part-based/local features. To perform part-based analysis, it is first required to extract keypoints/areas of interest from the signature images. The regions around these keypoints are then described using different descriptors. Hence, in the proposed approach in System 2, FAST [16] keypoint detector is used to detect keypoints in signature images. FAST keypoint detector is computationally efficient in comparison to well known keypoint detection methods, e.g., SIFT [17], Harris [18], and SURF [12] (we also compare results when we used SURF keypoint detector for finding the potential local areas of interest, in System 1). In addition, FAST gives a strong response on edges, which makes it suitable for the task of signature verification. Once the keypoints are detected, descriptor for each of the keypoints is computed using recently proposed part based descriptor, FREAK [19]. FREAK is a binary keypoint descriptor inspired by the retina in human visual system. These features are efficiently computed by sampling area around the keypoint on retinal pattern and encoding it as a binary string by comparing image intensities over this pattern. FREAK features are computationally very efficient in comparison to the well known part-based descriptors, i.e., SIFT [17] and SURF [12]. As the descriptors extracted using FREAK are binary, therefore Hamming distance is used for comparison of descriptors of query and reference signatures. The use of Hamming distance in-turn makes it computationally more efficient as it can be computed using a simple XOR operation on bit level. To categorize a signature as genuine, forged, or disguised, first it is binarized using the well known global binarization method OTSU [20]. We preferred using OTSU since we had fairly high resolution signature images and OTSU is also computationally efficient. After binarization, we applied the SURF/FAST keypoint detector on all the reference signatures, separately, to get the local areas of interest from these signatures. Then, obtained the descriptors of all of these keypoints present in all reference images using the FREAK keypoint descriptor. All of these keypoints and their associated descriptors describing important local information are added into a database. This resulted in a bag-of-features, which contained features for all of the keypoints which were collected from all reference signature images. Once the bag-of-features was created, keypoints and descriptors are extracted for the query/questioned signature. Now a comparison was made between the query signature keypoints and the keypoints present in our bag-of-features for that particular author. The same process of detecting local area of interest using FAST and then descriptors by FREAK is applied to the query image. After that we compared each of the query keypoints with the keyponts present in the bag-of-features. Kept this process going until all the query signatures keypoints were traversed. Finally, the average was calculated by considering the total number of query keypoints and the query keypoints matched with the bag-of-features. This represents the average local features of

the questioned signature that were present in the bag-of-features of that author. Now, if this average was greater than an empirically found threshold θ, (meaning, most of the questioned signature local features are matched with reference local features present in the bag-of-features), the questioned signature was classified as belonging to the authentic author, otherwise (meaning, there were only a few query keypoints for whom any match is found in the reference bag-of features), the query signature did not belong to the authentic author. Figure 1a shows an example reference (genuine) signature and Figure 1b shows the corresponding SURF-keypoints extracted from this reference (genuine) signature. Similarly Figure 1c shows a questioned (forged) signature and Figure 1d shows a questioned (disguised) signature, respectively, with SURF-keypoints extracted. Here blue dots represent the original questioned keypoints and red dots represent the keypoints matching with the reference signature keypoints.

3.2 System 3

In this system, given a scanned image as an input, first of all binarization is performed. Second, the image is normalized with respect to skew, writing width and baseline location. Normalization of the baseline location means that the body of the text line (the part which is located between the upper and the lower baselines), the ascender part (located above the upper baseline), and the descender part (below the lower baseline) is vertically scaled to a predefined size each. Writing width normalization is performed by a horizontal scaling operation, and its purpose is to scale the characters so that they have a predefined average width.

To extract the feature vectors from the normalized images, a sliding window approach is used. The width of the window is generally one pixel and nine geometrical features are computed at each window position. Thus an input text line is converted into a sequence of feature vectors in a 9-dimensional feature space. The nine features correspond to the following geometric quantities. The first three features are concerned with the overall distribution of the pixels in the sliding window. These are the average gray value of the pixels in the window, the center of gravity, and the second order moment in vertical direction. In addition to these global features, six local features describing specific points in the sliding window are used. These include the locations of the uppermost and lowermost black pixel and their positions and gradients, determined by using the neighboring windows. Feature number seven is the black to white transitions present within the entire window. Feature number eight is the number of black-white transitions between the uppermost and the lowermost pixel in an image column. Finally, the proportion of black pixels to the number of pixels between uppermost and lowermost pixels is used. For a detailed description of the features see [21].

Gaussian Mixture Models [22] have been used to model the handwriting of each person. More specifically, the distribution of feature vectors extracted from a person's handwriting is modeled by a Gaussian mixture density. For a

D-dimensional feature vector denoted as x, the mixture density for a given writer (with the corresponding model A) is defined as:

$$p(x\|A) = \sum_{i=1}^{m} w_i p_i(x)$$

In other words, the density is a weighted linear combination of M uni-modal Gaussian densities, $p_i(x)$, each parameterized by a $D \times 1$ mean vector, and D*D covariance matrix. For further details refer to [23], and [24].

4 Evaluation

4.1 Dataset

We used the test set of the 4NSigComp2010 signature verification competition for evaluations. This is the first ever publicly available dataset containing disguised signatures. The collection contains only offline signature samples. The signatures are collected by forensic handwriting examiners and scanned at 600 dpi resolution. The collection contains 125 signatures. There are 25 reference signatures by the same writer and 100 questioned signatures by various writers. The 100 questioned signatures comprise 3 genuine signatures written by the reference writer in her/his normal signature style and 7 disguised signatures written by the reference writer where s(he) tried to disguise herself/himself (the reference writer provided a set of signatures over a five day period); and 90 simulated signatures (written by 34 forgers freehand copying the signature characteristics of the reference writer. The forgers were volunteers and were either lay persons or calligraphers.). All writings were made using the same make of ball-point pen and using the same make of paper.

4.2 Results

As mentioned above, our evaluation data contained 3 genuine, 7 disguised, and 90 forged signatures. This is not a problem for the evaluation since we computed the Equal Error Rates (EER), calculated when the False Reject Rate (rate at which genuine and/or disguised signatures are misclassified as forged by a system) is same as the False Accept Rate (rate at which forged signatures are misclassified as genuine by a system).

We performed three experiments for evaluating the efficiency and EER of the three systems explained above and also compared their performance against all the other participants of the 4NSigComp2010 signature verification competition (the participants were tested on the same data under the same conditions in [4]).

- Experiment 1 that focused the classification of disguised, forged, and genuine signatures using FAST Kepypoint detector and FREAK features.
- Experiment 2 that focused the same classification using SURF Kepypoint detector and FREAK features.

Table 1. Summary of the comparisons performed among the participants of the 4NSigComp2010 (from systems 1 to 7) and mentioned local features based systems

System	FAR	FRR	EER	Time (sec.)
1	1.1	90	80	312
2	41.1	90	58	1944
3	20.0	70	70	85
4	0.0	80	70	19
5	13.3	80	55	45
6	87.0	10	60	730
7	1.1	80	70	65
(SURF-FREAK)	30	30	30	12
(FAST-FREAK)	30	30	30	**0.6**
(Bunke-GMM)	20	20	**20**	100

– Experiment 3 that focused the same classification using Bunke [21] features and GMM classification.

The results of these experiments are provided in Table 1. As shown in the table, all of our proposed local feature based systems, i.e., SURF-FREAK, FAST-FREAK, and Bunke-GMM outperform all the participants of the 4NSig Comp2010 signature verification competition in terms of EER. The best system from the competition could achieve an EER of 55%, whereas, the Bunke features based system reached an EER of 20% and both the SURF-FREAK and FAST-FREAK systems achieve an EER of 30%.

Furthermore, Table 1 also presents the performance comparison of the said systems on the basis of time. The time is given in seconds and is actually the average time taken by any algorithm to report its result on the authenticity of one questioned signature. For reporting this result, the system has to process the questioned as well as 25 reference signatures. Both of the proposed systems again outperformed all the participants. Specially the FAST-FREAK method is extremely time efficient. It succeeds from the other nine methods by a times of (520, 3240, 141, 31, 75, 1216, 108, 166, and 20 respectively). We performed all the tests at a machine with the following specifications.

– Processor: Intel Dual Core 1.73 GHz
– Memory: 1GB
– OS: WinXP Professional

A general drawback of most of the local features based approaches is the enormous amount of time they take to compute results. In our experiments, most of the participating systems were relying on global features except the proposed systems, yet the execution of the proposed systems was fairly time efficient than other systems. This shows that, if utilized properly, local feature approaches show the potential of improving both performance and efficiency of classification.

5 Conclusion and Future Work

In this paper we have compared three of our part based forensic signature verification systems with each other and also with all the systems that participated in the 4NSigComp2010 signature verification competition. We explicitly considered the time required and equal error rate achieved by each system. We performed three experiments with three different local feature sets. In the first system we used a combination of SURF-FREAK, in the second: a combination of FAST-FREAK, and in the third system we used the Bunke features. All of these local features based systems outperformed the participants of the 4NSigComp2010 signature verification competition by achieving EERs of 30%, 30%, and 20%, respectively. Whereas the EER of the best participant in the 4NSigComp2010 competition was 55%.

Furthermore, we have made a time efficiency comparison among the considered local features based systems and the participants of the 4NSigComp2010. Here again our local features based systems out-performed other participants and one of our systems (i.e., FAST-FREAK) outperformed all other systems remarkably.

In future we plan to use larger data sets where disguised signatures from large number of authors are present in the test set. Regarding the systems' outcomes, we plan to enable them produce likelihood ratios according to Bayesian approach, which will make these systems even more useful in the real world forensic casework. This, however, is a difficult task since respective likelihood computation of multiple classes is required in this case.

References

1. Plamondon, R., Lorette, G.: Automatic signature verification and writer identification – the state of the art. Pattern Recognition 22, 107–131 (1989)
2. Plamondon, R., Srihari, S.N.: On-line and off-line handwriting recognition: A comprehensive survey. IEEE TPAMI 22, 63–84 (2000)
3. Impedovo, D., Pirlo, G.: Automatic Signature Verification: The State of the Art. IEEE Trans. on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 38, 609–635 (2008)
4. Liwicki, M., van den Heuvel, C.E., Found, B., Malik, M.I.: Forensic signature verification competition 4NSigComp2010 - detection of simulated and disguised signatures. In: ICFHR 2010, pp. 715–720 (2010)
5. Malik, M.I., Liwicki, M., Dengel, A.: Evaluation of local and global features for offline signature verification. In: 1st Int. Workshop on Automated Forensic Handwriting Analysis (AFHA), pp. 26–30 (2011)
6. De Stefano, C., Marcelli, A., Rendina, M.: Disguising writers identification: an experimental study. In: IGS 2009, pp. 99–102 (2009)
7. Fiel, S., Sablatnig, R.: Writer Retrieval and Writer Identification Using Local Features. In: DAS 2012, pp. 145–149 (2012)
8. Zhiyi, Z., Lianwen, J., Kai, D., Xue, G.: Character-SIFT: A Novel Feature for Offline Handwritten Chinese Character Recognition. In: ICDAR, pp. 763–767 (2009)

9. Jin, Z., Qi, K.-Y., Chen, K.: SSIFT: An Improved SIFT Descriptor for Chinese Character Recognition in Complex Images. In: CNMT, pp. 1–5 (2009)
10. Song, W., Uchida, S., Liwicki, M.: Comparative Study of Part-Based Handwritten Character Recognition Methods. In: ICDAR 2011, pp. 814–818 (2011)
11. Ta, D.-N., Chen, W.-C., Gelfand, N., Pulli, K.: SURFTrac: Efficient tracking and continuous object recognition using local feature descriptors. In: IEEE C. S. Conf. on Computer Vision and Pattern Recognition, pp. 2937–2944 (2009)
12. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-Up Robust Features (SURF). Comput. Vis. Image Underst. 110(3), 346–359 (2008)
13. Jeong, K., Moon, H.: Object Detection Using FAST Corner Detector Based on Smartphone Platforms. In: First Int. Conf. on Computers, Networks, Systems and Industrial Engineering (CNSI), pp. 111–115 (2011)
14. Bilinski, P., Bremond, F., Kaaniche, M.B.: Multiple object tracking with occlusions using HOG descriptors and multi resolution images. In: ICDP 2009, pp. 1–6 (2009)
15. Diem, M., Sablatnig, R.: Recognition of Degraded Handwritten Characters Using Local Features. In: ICDAR 2009, pp. 221–225 (2009)
16. Rosten, E., Drummond, T.: Fusing points and lines for high performance tracking. In: 10th Int. Conf. on Computer Vision, pp. 1508–1515 (2005)
17. Lowe, D.G.: Object recognition from local scale-invariant features. In: 7th Int. Conf. on Computer Vision, pp. 1150–1157 (1999)
18. Harris, C., Stephens, M.: A combined corner and edge detector. In: 4th Alvey Vision Conf., pp. 147–151 (1988)
19. Alahi, A., Ortiz, R., Vandergheynst, P.: FREAK: Fast Retina Keypoint. In: CVPR 2012, pp. 510–517 (2012)
20. Otsu, N.: A threshold selection method from gray-level histograms. Automatica 111, 285–296 (1975)
21. Marti, U.-V., Bunke, H.: Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system. IJPRAI 110(15), 65–90 (2001)
22. Marithoz, J., Bengio, S.: A comparative study of adaptation methods for speaker verification. In: Int. Conf. on Spoken Language Processing, pp. 581–584 (2002)
23. Liwicki, M.: Evaluation of Novel Features and Different Models for Online Signature Verification in a Real-World Scenario. In: 14th Conf. of Int. Graphonomics Society, pp. 22–25 (2009)
24. Schlapbach, A., Liwicki, M., Bunke, H.: A Writer Identification System for On-line Whiteboard Data. PR 41(7), 2381–2397 (2008)