Fingerprint Image Invariant Feature Extraction Algorithm

Jinghong Xu[1,a], Xinyou Dong[2,b]

[1]Dept. Computer Science and Technology, Shanghai Jianqiao University, Shanghai, China

[2]Dept. Algorithm Design, ZTE, Shanghai, China

[a]xujinghong@gench.edu.cn, [b]xinyoudong@gmail.com

Keywords: fingerprint authentication, local features, MSER, detectors, descriptors

Abstract: Fingerprint based authentication systems play a vital role in identifying an individual. The existing systems depend on specific feature points. Designing a reliable fingerprint authentication system is very challenging, since not all fingerprint information is available. Further, the information obtained is not always accurate due to cuts, scars, sweat, distortion and various skin conditions. Moreover, feature detection and description algorithms are typically computationally intensive, which prevents them from achieving the speed of sight real-time performance. In addition, algorithms differ in their capabilities and some may favor and work better given a specific type of input compared to others. As such, it is essential to compactly report their pros and cons as well as their performances. This paper provides a comprehensive overview on the state-of-the-art and recent advances in feature detection and description algorithms. It compares, reports and discusses their performance and capabilities. And then the Maximally Stable Extremal Regions algorithm is selected to extract the fingerprint features. The result shows that the feature points of fingerprint image are rotation, scale and affine invariant.

1. Introduction

Biometrics refers to metrics of human characteristics is for body measurements and calculations. The goal of biometric identity verification system is to recognize individuals. The features which are used by verification system can be categorized into:

- anatomical characteristics: fingerprints, signature, face, DNA, finger shape, hand geometry, iris, retina, ear;
- behavioral characteristics: typing rhythm, gait, gestures, and voice[1];

Biometric identifiers can be compared on the following factors: universality, distinctiveness, permanence, collectability, performance, acceptability and circumvention [1, 2]. Because of the well-known distinctiveness (individuality) and persistence properties of fingerprints as well as cost and maturity of products, fingerprint recognition systems are one of the most deployed characteristic and extensively used biometric systems[3].

Fingerprints are comprised of two direction-oriented patterns, ridges and valleys. Practically, fingerprint recognition techniques can be coarsely classified into four categories:

- correlation-based[3];
- ridge feature-based[3];
- local fingerprint image reconstruction based[4];
- minutiae-based recognition techniques[3].

In correlation based recognition, two images are superimposed and the correlation between corresponding pixels for different alignments provides the similarity between the two fingerprint images. This produces inaccurate results due to its dependency on properties of the image, undesirable changes of global structure, and high computational cost. In ridge feature based techniques, the ridge feature in form of a finger code is used. A similarity score is obtained by computing the difference of two finger code vectors[5]. The downside of a ridge-feature based technique is that, alignment of the fingerprint images is essential[5]. The minutiae-based technique locates the minutiae in the given finger, as well as the template and tries to match using minutiae's relative position[6]. This technique has many drawbacks such as,

- extracting minutiae from poor quality fingerprints;
- fingers with cuts and bruises provide images with broken ridges thereby leading to false minutiae detection[7];
- the ridge patterns which can be useful to improve the accuracy is ignored[8];
- the overlapping region is minimal between the input and template fingerprint. Moreover, anecdotal evidence shows that a small number of minutiae are present in a fraction of the population, which causes vulnerability in this system[8].

Local feature[9, 10] reflects different aspects of information embedded in a local image patch. Moreover, they are independent of each other and none of them can be covered by the others. They can provide complementary discriminating power to each other for recognition biometric images. Thus, better recognition performance could be expected using local features.

This paper provides a comprehensive overview on the state-of-the-art and recent advances in local feature detection algorithms. Specifically, the paper starts by overviewing fundamental concepts that constitute the core of feature detection and description algorithms. It then compares, reports and discusses their performance and capabilities. The Maximally Stable Extremal Regions (MSER) algorithm[11] being the best one to choose.

The rest of the paper is organized as follows. Section II provides an overview of the recent state-of-the-art feature detection and description algorithms proposed in literature. It also summaries and compares their performance and accuracy under various transformations. In Section III, the MSER algorithm is studied in detail in terms of their recent derivatives. And then, using the most fitful MSER algorithm to calculate the local fingerprint feature points. Finally, Section V concludes the paper with outlooks into future work.

2. Definitions and principles

This section describes a set of feature in an image. It also summarizes the metrics used to measure the quality of the generated feature descriptors.

2.1 Local Features

Local image features can be defined as a specific pattern which unique from its immediately close pixels[9, 10]. Such properties include edges, corners, regions, etc. Indeed, these local features represent essential anchor points that can summarize the content of the frame (with the aid of feature descriptors) while searching an image. These local features are then converted into numerical descriptors, representing unique and compact summarization of these local features. Local features provide a powerful tool that can be used in a wide range of computer vision and robotics applications, such as real-time visual surveillance, image retrieval, video mining, object tracking, mosaicking, target detection, and wide baseline matching to name few[12]. To illustrate on the usefulness of such local features, consider the following example. Given an aerial image, a detected edge can represent a street, corners may be street junctions, and homogeneous regions can represent cars, roundabouts or buildings.

The term detector traditionally refers to the algorithm or technique that detects (or extracts) these local features and prepare them to be passed to another processing stage that describe their contents, i.e. a feature descriptor algorithm. That is, feature extraction plays the role of an intermediate image processing stage between different computer vision algorithms. In this work, the terms detector and extractor are interchangeably used.

2.2 Ideal Local Features

In general, a local feature typically has a spatial extent which is due to its local pixels neighborhood. That is, they represent a subset of the frame that is semantically meaningful, e.g. correspond to an object (or a part of an object). Ultimately, it is infeasible to localize all such features as this will require the prerequisite of high-level frame (scene) understanding[9]. As such, those features detection algorithms tries to locate these features directly based on the intensity patterns in the input frame. The selection of these local features can indeed greatly impact the overall system performance[10].

Intuitively, a given computer vision applications may favor one quality over another [9]. Repeatability, arguably the most important quality, is directly dependent on the other qualities (that is, improving one will equally improve repeatability). Nevertheless, regarding the other qualities, compromises typically need to be made. For example, distinctiveness and locality are competing properties (the more local a feature, the less distinctive it becomes, making feature matching more difficult). Efficiency and quantity are another example of such competing qualities. A highly dense features are likely to improve the object/scene recognition task, but this, however, will negatively impact the computation time. The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

2.3 Feature Detectors

The technical literature is rich with features detections algorithms. However, no ideal detector exists until today. This is mainly due to the virtually infinite number of possible computer vision applications (that may require one or multiple features), the divergence of imaging conditions (changes in scale, viewpoint, illumination and contrast, image quality, compression, etc.) and possible scenes. The computational efficiency of such detectors becomes even more important when considered for real-time applications.

The most important local features include: Edges, Corners, Regions. One can intuitively note that there is a strong correlation between these local features. For example, multiple edges sometimes surround a region, i.e. tracking the edges defines the region boundaries. Similarly, the intersection of edges defines the corners[13]. A summary for the well-known feature detectors can be found in table I. The performance of many of the state-of-the-art detectors is compared in table II.

Table 1 A summary of the state-of-the-art feature detectors[10]

Category	***Classification***	***Methods and Algorithms***
Edge based	Differentiation based	Sobel, Canny
Corner based	Gradient based	Harris (and its derivatives), KLT, Shi-Tomasi, LOCOCO, SLOCOCO
Corner based	Template based	FAST, AGAST, BRIEF, SUSAN, FASTER
Corner based	Contour based	ANDD, DoG-curve, ACJ, Hyperbola fitting, etc.
Corner based	Learning based	NMX, BEL, Pb, MS-Pb, gPb, SCG, SE, tPb, DSC, Sketch Tokens, etc.
Blob (interest point)	PDE based	SIFT (and its derivatives), SURF (and its derivatives), CenSurE, LoG, DoG, DoH, Hessian (and its derivatives), RLOG, MO-GP, DART, KAZE, A-KAZE, WADE, etc.
Blob (key point)	Template based	ORB, BRISK, FREAK
Blob (interest region)	Segmentation based	MSER (and its derivatives), IBR, Salient Regions, EBR, Beta-Stable, MFD, FLOG, BPLR

As was reported in many performance comparison surveys in the computer vision literature[9, 14, 15], the MSER[11] algorithms have shown an excellent performance in terms of the invariance and other feature qualities (see table II). Due to these facts, the MSER algorithms was extended to several derivatives with different enhancements (that will be reported on later sections). As such, the following section of this paper considers reporting the algorithmic standard and derivatives of the MSER algorithms.

Table 2 A summary of the performance of dominant features detection algorithms[9]

Features Detector rotation	Rotation	Scale	Affine
Harris	+	-	-
Hessian	+	-	-
SUSAN	+	-	-
Harris-Laplace	+	+	-
Hessian-Laplace	+	+	-
DoG	+	+	-
Salient Regions	+	+	-
SURF	+	+	-
SIFT	+	+	-
MSER	+	+	+

3. Mathematical models

3.1 Standard Maximally Stable Extremal Regions

The MSER detector can be informally described as follows [11]: Assume an initially empty grid that corresponds to an intensity image. We start inserting all pixels of intensity of value 0 to their corresponding locations, then all pixels of intensity value of 1, 2, and so on, until all pixels are reinserted into their corresponding locations, and the image is completely restored. Equivalently, the intensity image is continuously thresholded starting with threshold 0 up to 255 with a threshold increment. At each threshold, all pixels with values that fall below the current threshold are painted white and the remaining pixels are painted black. As the threshold is increasing, some white regions will show, some will merge, until ultimately all regions will merge into a single large one. In this process, we keep monitoring the size of each white region, i.e. its cardinality , as a function of the threshold value t. Then, an MSER is detected if has a local minimum, where:

(1)

In this case, the detected MSERs correspond to the bright regions. For dark MSERs, the inverted intensity image is used instead.

The formal definition of the MSERs[11] is as follows:

Definition 1: Let be a sequence of nested extremal regions, i.e. .

Extremal region is maximally stable if has a local minimum at t, where is the threshold increment.

The word 'extremal' refers to the property that all pixels inside the MSER have either higher (bright extremal regions) or lower (dark extremal regions) intensity than all the pixels on its outer boundary.

The MSER is controlled by four main parameters, namely the threshold increment , the minimum and maximum size of each region, and the maximum area variation defined by the stability function . There are no optimal values for these four parameters. The lower the value of , the more accurate (but the slower) the algorithm becomes. Typically, is selected in the range of 4–7.

3.2 MSER Derivatives

Maximally stable extremal regions (MSER) algorithm was proposed by Matas et al in 2002. Since then number of region detection algorithms have been proposed based on the MSER technique. The following is a list of five MSER derivatives presented in chronological order.

3.2.1 N-Dimensional Extension

The algorithm was extended first in 2006 for 3D segmentation [16] by extending the neighborhoods search and stability criteria to 3D image data instead of 2D intensity date. Later on, in 2007, another extension for N-dimensional data space was proposed by Vedaldi in [17], and later on the same year, an extension to vector-valued function that can be exploited with the three-color channels was also provided in [18].

3.2.2 Linear-Time MSER Algorithm

In 2008, Nister and Stewenius proposed a new processing flow that emulates real flood-filling in [19]. The new linear-time MSER algorithm has several advantages over the standard algorithm such as the better cache locality, linear complexity, etc. An initial hardware design was proposed in [20].

3.2.3 The Extended MSER(X-MSER) Algorithm

The standard MSER algorithm searches for extremal regions from the input intensity frame only. However, in 2015, the authors of [21] proposed an extension to the depth (space) domain noting out the correlation between the depth images and intensity images, and introduced the extended MSER detector, which was patented in [22].

3.2.4 The Parallel MSER Algorithm

One of the major drawbacks of the MSER algorithm is the need to run it twice on every frame to detect both dark and bright extremal regions. To circumvent on these issues, the authors proposed a parallel MSER algorithm [23]. Parallel in this context refers to the capability of detecting both extremal regions in a single run. This algorithmic enhancement showed great advantages over the standard MSER algorithm such as a considerable reduction in the execution time, required hardware resources and power, etc. This parallel MSER algorithm has few US patents that are associated with it (e.g. [24]).

3.2.5 Other MSER Derivatives

Other algorithms that were inspired from the MSER algorithm include the Extremal Regions of the Extremal Levels [25, 26] algorithm and the Tree-based Morse Regions (TBMR)[27].

4. Mser Detection Result

Fig.1 Results by MSER

In our project, proposed fingerprint scheme purposes on supporting the search for a large scale of multimedia data with tens or hundreds billions of fingerprint. Efficiency and accuracy are the key indicator to the selected algorithm.

We have evaluated our algorithm for the task of fingerprint matching and tracking in a dataset of LivDet. Figure 1 shows MSER feature points of fingerprint image. And the result proves that the MSER algorithm is rotation, scale and affine invariant.

5. Conclusion and future workmser detection result

It presents an overview of the recent state-of-the-art algorithms proposed in literature. It starts by reviewing basic yet fundamental concepts that are related to these algorithms. It also provided a brief comparison on their performance and capabilities based on different metrics. The algorithms have been compared in terms of quality of the extracted features under image transformation exist in real-life applications, such as image rotation, scaling and affine. From this class of algorithms, the most frequently used algorithms MSER is selected for detail exploration with its algorithmic derivatives.

As future work, further optimization of the fingerprint recognition MSER algorithm would improve the system. After that, we would use local feature points to comprise the fingerprint sketch to search the large scale database.

Acknowledgment

This project was supported by the fingerprint recognition research projects, KYJF15BB15005, granted by Shanghai Jianqiao University.

References

[1] D. Maltoni, D. Maio, A. Jain, and S. Prabhakar, Handbook of fingerprint recognition: Springer Science & Business Media, 2009.

[2] M. Gamassi, M. Lazzaroni, M. Misino, V. Piuri, D. Sana, and F. Scotti, "Accuracy and performance of biometric systems," in Instrumentation and Measurement Technology Conference, 2004. IMTC 04. Proceedings of the 21st IEEE, 2004, pp. 510-515.

[3] Y. Song, C. Lee, and J. Kim, "A new scheme for touchless fingerprint recognition system," in Intelligent Signal Processing and Communication Systems, 2004. ISPACS 2004. Proceedings of 2004 International Symposium on, 2004, pp. 524-527.

[4] S. Bakhtiari, S. S. Agaian, and M. Jamshidi, "Local fingerprint image reconstruction based on gabor filtering," in SPIE Defense, Security, and Sensing, 2012, pp. 840602-840602-11.

[5] (2016). Ridge feature-based matching | Griaule Biometrics. Available: http://www.griaulebiometrics.com/en-us/book/understandingbiometrics/types/matching/feature-based.

[6] X. Luo, J. Tian, and Y. Wu, "A minutiae matching algorithm in fingerprint verification," in Pattern Recognition, 2000. Proceedings. 15th International Conference on, 2000, pp. 833-836.

[7] R. Zhou, D. Zhong, and J. Han, "Fingerprint identification using SIFTbased minutia descriptors and improved all descriptor-pair matching," Sensors, vol. 13, pp. 3142-3156, 2013.

[8] U. Park, S. Pankanti, and A. Jain, "Fingerprint verification using SIFT features," in SPIE Defense and Security Symposium, 2008, pp. 69440K-69440K-9.

[9] T. Tuytelaars, K. Mikolajczyk, et al., “Local invariant feature detectors: a survey,” Foundations and trends in computer graphics and vision, vol. 3, no. 3, pp. 177–280, 2008.

[10] Y. Li, S. Wang, Q. Tian, and X. Ding, “A survey of recent advances in visual feature detection,” Neurocomputing, vol. 149, pp. 736–751, 2015.

[11] J. Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide-baseline stereo from maximally stable extremal regions,” in Proceedings of the British Machine Vision Conference, pp. 384–393, 2002.

[12] S. Krig, “Computer vision metrics, survey, taxonomy, and analysis. Apress,” 2014.

[13] Q. Liu, R. Li, H. Hu, and D. Gu, “Extracting semantic information from visual data: A survey,” Robotics, vol. 5, no. 1, p. 8, 2016.

[14] K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” International journal of computer vision, vol. 60, no. 1, pp. 63–86, 2004.

[15] O. Miksik and K. Mikolajczyk, “Evaluation of local detectors and descriptors for fast feature matching,” in Pattern Recognition (ICPR), 2012 21st International Conference on, pp. 2681–2684, IEEE, 2012.

[16] M. Donoser and H. Bischof, “3d segmentation by maximally stable volumes (msvs),” in Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, vol. 1, pp. 63–66, IEEE, 2006.

[17] Vedaldi, “An implementation of multi-dimensional maximally stable extremal regions,” Feb, vol. 7, pp. 1–7, 2007.

[18] P.-E. Forssen, “Maximally stable colour regions for recognition and matching,” in Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on, pp. 1–8, IEEE, 2007.

[19] D. Nister and H. Stew ´ enius, “Linear time maximally stable extremal regions,” in European Conference on Computer Vision, pp. 183–196, Springer, 2008.

[20] S. Alyammahi, E. Salahat, H. Saleh, and A. Sluzek, “A hardware accelerator for real-time extraction of the linear-time mser algorithm,” in Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE, pp. 65–69, IEEE, 2015.

[21] Salahat, H. Saleh, S. Salahat, A. Sluzek, M. Al-Qutayri, and M. Ismail, “Extended mser detection,” in IEEE International Symposium on Industrial Electronics, Rio de Janeiro, Brazil, 3-5 Jun. 2015.

[22] E. N. Salahat, H. H. M. Saleh, S. N. Salahat, A. S. Sluzek, M. AlQutayri, B. Mohammad, and M. I. Elnaggar, “Object detection and tracking using depth data,” Oct. 23 2014. US Patent App. 14/522,524.

[23] E. Salahat, H. Saleh, A. Sluzek, M. Al-Qutayri, B. Mohammad, and M. Ismail, “A maximally stable extremal regions system-on-chip for real-time visual surveillance,” in Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE, pp. 002812–002815, IEEE, 2015.

[24] E. N. Salahat, H. H. M. Saleh, A. S. Sluzek, M. Al-Qutayri, B. Mohammad, and M. I. Elnaggar, “Architecture and method for real-time parallel detection and extraction of maximally stable extremal regions (msers),” Apr. 12 2016. US Patent 9,311,555.

[25] M. Faraji, J. Shanbehzadeh, K. Nasrollahi, and T. B. Moeslund, “Erel: extremal regions of extremum levels,” in Image Processing (ICIP), 2015 IEEE International Conference on, pp. 681–685, IEEE, 2015.

[26] M. Faraji, J. Shanbehzadeh, K. Nasrollahi, and T. B. Moeslund, “Extremal regions detection guided by maxima of gradient magnitude,” IEEE Transactions on Image Processing, vol. 24, no. 12, pp. 5401–5415, 2015.

[27] Y. Xu, P. Monasse, T. Geraud, and L. Najman, “Tree-based morse regions: A topological approach to local feature detection,” IEEE Transactions on Image Processing, vol. 23, no. 12, pp. 5612–5625, 2014.