International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 9 (2017) pp. 1935-1942

Minutiae-based Matching Scheme for Fingerprint Identification using the Grassfire Spot Matching

Dae-Seong Jeoune
Associate Professor,
Dept. of Drone Photo & Videography, Daegu Mirae College
Gyeongsan, Gyeongbuk, Republic of Korea.
ORCID: 0000-0003-1669-5700

Chan-Myeong Han
CEO, EO3 Co. Ltd.
Research Institute, M+VISION Co. Ltd.
Daegu Metropolitan City, Republic of Korea.

Yun-Kyoo Ryoo
Associate Professor,
Dep't of Medical Computer Science, Daegu Health College
Daegu Metropolitan City, Republic of Korea.

Jae Hak Sa
Director,
MIS Division, CRETEC
Daegu Metropolitan City, Republic of Korea .

Seunghui Cha
Dep't of Computer Engineering
Yeungnam University
Gyeongsan, Gyeongbuk, Republic of Korea

Wook Hyun Kim*
Professor,
Dep't of Computer Engineering, Yeungnam University
Gyeongsan, Gyeongbuk, Republic of Korea.

Abstract
As for the grassfire spot matching algorithm, it has been originally proposed for spot matching in 2-DE. There exists many things in common between spot matching in 2-DE and minutiae-based fingerprint matching. So, the grassfire algorithm is supposed to be applied and work well for minutiae-base fingerprint matching. In this paper, the grassfire spot matching algorithm is described in detail with several literatures related to spot matching in 2-DE. An experiment is performed to present algorithm's potential possibilities for the field of minutiae-based fingerprint matching. Only position information is used to match minutiae. From the result, nonetheless, we can find out it shows great outcome. The future research includes topics to elaborate on matching accuracy and speed by considering other features such as type and direction of minutiae as well as position information.

Keywords: Biometric Recognition, Fingerprint, Minutiae, Grassfire Spot Matching, 5-NNG

INTRODUCTION
Fingerprint matching is the most popular biometric technique used in automatic personal identification. Law enforcement agencies use it routinely for criminal identification. It is also being used in several other applications such as access control for high security installations, credit card usage verification and employee identification [1]. The increased emphasis on the privacy and security of information stored in various databases, automatic personal identification has become a very important topic with the advent of electronic banking, e-commerce and smartcards [2]. The main reason for the popular use of fingerprints as a means for identification is that the fingerprint of an individual person is unique and remains invariant [3].

Fingerprint matching is the most crucial step in a series of steps for fingerprint identification. Especially, minutiae-based fingerprint matching is the most active research field. In previous works, however, only old point matching algorithms are being tried for fingerprint identification.

The grassfire matching algorithm is a recently developed point matching algorithm and considered to be very effective for such point matching problems. Spot matching in 2-DE and fingerprint matching based on minutiae have so much things in common. Although the grassfire spot matching algorithm was originally invented for spot matching in 2-DE, it is thought to be applied successfully in fingerprint matching for this reason.

In this paper, we describe how similar problems exist between them. Also, the grassfire matching algorithm is explained in detail to show the way it can be applied to fingerprint matching. Finally, the grassfire matching algorithm is verified its' effectiveness through a very simple experiment, showing that it gives great possibilities for fingerprint matching.

RELATED WORKS
A. Fingerprint for Biometric Recognition
A Fingerprint represents the image of the surface of the skin of the fingertip. A typical structure of a fingerprint consists of ridges(black lines) separated by valleys. The ridge pattern in a fingerprint can be described as an oriented texture pattern with fixed dominant spatial frequency and orientation in a local neighborhood. The frequency is dependent on inter-ridge spacing and orientation on flow pattern exhibited by the ridges. The global pattern of fingerprint is used to determine the class [4]. Region of a fingerprint where the ridge pattern makes it visually prominent are called singularities [5]. There are two types of fingerprint singularities: core and delta. They are very useful for determining fingerprints class.

A closer analysis of the fingerprint reveals some anomalies of the ridges, such as ridge endings, bifurcations, crossovers, short ridges etc. These local features of fingerprints called minutiae can be used by manual or automatic fingerprint identification. The most important features are ridge ending and ridge bifurcation. These basic features of fingerprint - singularities and minutiae - are shown in figure 1 [6].

Figure 1. Basic Features of Fingerprint

Ridge ending is the abrupt end of a ridge. Ridge bifurcation is a single ridge that divides into two ridges. Short ridge is a ridge that commences, travels a short distance and then ends. Island is a single small ridge inside a short ridge or ridge ending that is not connected to all other ridges. Ridge enclosure is a single ridge that bifurcates and reunites shortly afterward to continue as a single ridge. Spur is a bifurcation which a short ridge branching off a long ridge. Crossover is a short ridge that runs between two parallel ridges. Delta is a Y-shaped ridge meeting. Core is a U-tern in the ridge pattern [7].

Automated fingerprint recognition based on minutiae includes several processes such as Fingerprint Acquisition, Fingerprint Segmentation, Fingerprint Image Enhancement, Feature Extraction, and Minutiae Matching [8]. Fingerprint Acquisition is the process of obtaining fingerprint images from fingerprint scanners. Fingerprints can be sensed using numerous technologies. The traditional "ink and paper" method, still used by many law enforcement agencies, involves applying ink to the finger surface, rolling the finger from one side of the nail to the other on a card and finally scanning the card to generate a digital image.

In the more popular live-scan method, a digital image is directly obtained by placing the finger on the surface of a fingerprint reader. Optical sensors based on the frustrated total internal reflection(FTIR) technique are commonly used to capture live-scan fingerprints in forensic and government applications, while solid-state touch and sweep sensors-silicon-based devices that measure the differences in physical properties such as capacitance or conductance of the friction ridges and valleys – dominate in commercial applications [9].

Fingerprint Segmentation is the process of separating foreground from noisy background without excluding ridge-valley regions. And it does not include meaningless background. Image Enhancement includes keeping the original ridge flow pattern without altering the singularity, joining broken ridges and cleaning artifacts between pseudo-parallel ridges. Feature extraction is to locate and classify minutiae points from the enhanced image. Minutiae Matching is to compare two fingerprint images based on minutiae points to check whether two fingerprints are identical or not [8].

B. Fingerprint Matching based on Minutiae

Mital and Teoh proposed a minutiae-based matching algorithm [3], which uses five minutiae of its closest neighbors to form geometric structure along with a scoring method. Their algorithm is simple, very effective, and rotationally invariant. However, it has the following two weak points. First, their algorithm becomes sensitive to false minutiae because a false minutia is chosen, if it is closer to true ones, for the structure construction. Second, their scoring method penalizes too heavily discrepancy in the minutiae type between a query and a template even though the geometric structures are the same or similar.

In the literature [10], a new fingerprint matching algorithm is proposed for locally deformed fingerprints based on a geometric structure of minutiae, called the radial structure, which is a collection of lines from minutiae that connect its Voronoi neighbors. Ning Liu et al. propose a fingerprint matching algorithm based on DT net [11]. It uses DT in fingerprint matching and then develops a matching algorithm based on DT net to find reference minutiae pairs(RMPs). Using DT on the topological structure of minutiae set, a DT net is formed with minutiae as vertexes. From the nets of the input minutiae set and template minutiae set, select out a certain pairs of minutiae which have similar structures as RMPs for aligning, and matching is carried out based on point pattern. The experiment is conducted on FVC2002, and the result indicates the validity of our algorithm.

Chen et al. proposed an algorithm to use minutiae for fingerprint recognition, in which the fingerprint's orientation field is reconstructed from minutiae and further utilized in the matching stage to enhance the system's performance. First, they have produced "virtual" minutiae by using interpolation in the sparse area, and then used an orientation model to reconstruct the orientation field from all "real" and "virtual" minutiae. A decision fusion scheme is used to combine the reconstructed orientation field matching with the conventional minutiae-based matching [12].

Min and Thein have presented a recognition system which combines both the statistical and geometry approaches. The core point(CP) of the input fingerprint is detected and located in the canter. Then, the fingerprint image is cropped around the based point. Fingerprint features such as minutiae points' determination, their coordinates location, and radius of arcs for each ridge are stored in different databases. For a testing fingerprint image, the features are compared with these pre-defined databases and the decision is made by a voting system [13].

In Wei-bo et al. [14] each minutia was defined by the type and the relative topological relationship among the minutia and its 5 nearest neighbors. Qi et al. [15] proposed a fingerprint matching algorithm using the elaborate combination of minutiae and curvature maps from fingerprint images. First, they computed the curvature in a simple way based on orientation field, and then performed the sampling operation on the curvature map around each minutia to get

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 9 (2017) pp. 1935-1942

the fixed length minutiae specifiers. Second, a similarity measurement was defined between two specifiers. Third, they found the reference points pair based on computing the least squared error of Euclidean distance between these two specifiers. Finally, they completed the matching task by aligning the two fingerprint minutiae sets and accounting the number of overlapping minutiae.

MATERIALS AND METHODOLOGY

A. Spot Matching using Topological Patterns(SMTP)

Human can match patterns better than computers. Considering how to match patterns by human is very important because a hint for spot matching algorithms can be taken from human perception. Humans match spots using patterns. Points construct unique patterns and humans match points by comparing similarity of patterns of points. Point can have a variety of traits such as color, shape, size and non-visual traits. Point patterns expressed only by two-dimensional positional traits for points are called topological patterns. Relationship between the center spot and neighbor spots can be described mathematically by graph theory [16]. Topological patterns are subgraphs consisting of a center spot and its neighbor spots. Neighbor spots can also be defined using graph theories.

There are many graph theories available such as relative neighbor graph, Gabriel graph, k-nearest neighbor graph(k-NNG) and Delaunay graph. It is very important to select one of these graph theories. The results can heavily depend on a graph to be used. The literature [16] used 5-NNG graph in spot matching of 2-DE after conducting a series of experiments for graphs. It compares two topological patterns to determine whether two spots from different two gels are same spots as humans do. A topological pattern consists of the center point and its neighbor points. Neighbor points are top 5 nearest points from the center point in 5-NNG graph.

By intuition, human can compare two topological patterns without any efforts. An algorithm can be described mathematically to simulate human's behavior. The easiest way of comparing two patterns is to overlap two patterns exactly by their center spots as in figure .2. Point patterns from two different sources have different geometric parameters such as transposition, scale and rotation. They should be corrected before comparing two patterns. Similarity transform can be used to align patterns in two-dimensional space.

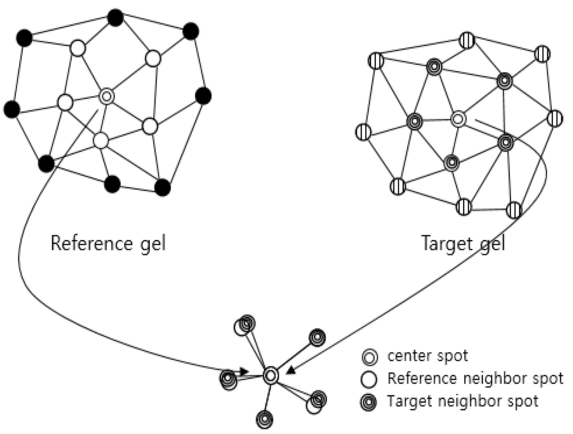


Figure 2. Spot Matching by Topological Patterns

In the figure 3, two subgraphs are superimposed with central spots as the center in the process of similarity transform as P1(Transition). A pair of spots other than the center pair is selected and one of the two subgraphs is rotated to adjust the rotation disparity as P2(Rotation). The pair used in aligning two subgraphs is called "pivot pair". Finally a scale factor is applied to adjust the size as P3(Scaling) in the figure 3. Once the geometric disparities are removed, neighbor spots are matched by distance. Two nearest spots become a pair. There might be unmatched spots if certain thresholds for distance and rotation are applied. The degree of similarity should be calculated to determine whether the two center spots are matched or not. In the literature [12], three factors such as the number of matched spot pairs(MP), the number of unmatched spots(UMP) and the normalized Hausdorff distance(NHD) should be considered to get the best matched pair.

There are two point sets in 2-DE, which are reference gel and target gel. Reference gel is pre-registered point set and target gel is one to be tested or matched. One point from target gel is matched with every point from reference gel as described above. Only one pair with the highest similarity is determined as "a matched pair". Spot matching algorithm using topological patterns formed by the center spot and its neighbor spots is called "spot matching by topological patterns(SMTP)".

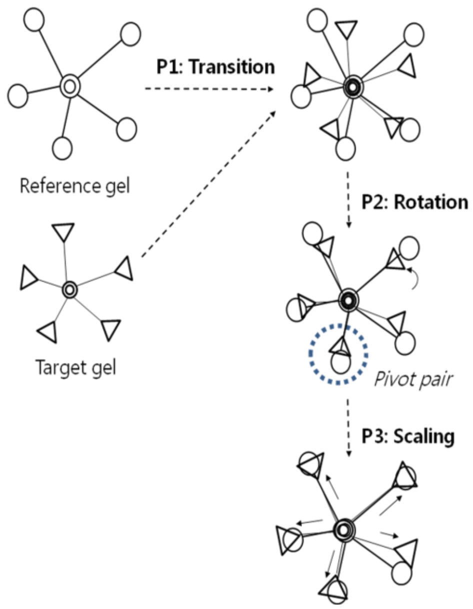


Figure 3. Similarity Transform

The whole process for SMTP is described in the figure 4. Points from reference and target gel are extracted using spot detection algorithms as line 1 and 2. For spot pair (ts, rs), two subgraphs are extracted using 5-Nearest Neighbor Graph as line 4 and 5. Spot matching is performed for every combination of neighbor spots, (nts, nrs) and the best matching result among them are chosen for the final matching result as line 8 to line 16. The best pivot pair and the best spot pair can be judged with the number of matched pairs, the number of unmatched spots and normalized Hausdorff distance. The number of matched pairs is prioritized the most. The number of unmatched spots is followed by it and the normalized Hausdorff distance takes the least priority. This rule is well described in line 12 to 22. For one spot of target gel, all of reference spots are matched respectively and the best match with the best result among them is chosen for the final match.

```
01: RefPoints ← ExtractPoint(ref_image)
02: TarPoints ← ExtractPoint (tar_image)
03: for each spot ts∈TarPoints do
04:   NTS ← get_neighbor_spots (ts, 5NNG)
05:   for each spot rs∈RefPoints do
06:    NRS ← get_neighbor_spots (rs, 5NNG)
07:    BSM ← NULL
08:    for each neighbor spot nrs∈NRS do
09:     for each neighbor spot nts∈NTS do
10:       NTST ← similarity_transform (NRS, NTS, nrs, nts)
11:        (MP, UMP, NHD) ← spot_matching (NRS, NTS,
           maxAngle, maxDist)
12:       if BestMP < MP then
13:         (BestMP, BestUMP, BestNHD) ← (MP, UMP, NHD)
14:       else if BestMP = MP then
15:         if BestUMP > UMP then
16:           (BestMP, BestUMP, BestNHD) ← (MP, UMP, NHD)
17:         else if BestUMP = UMP then
18:          if BestNHD > NHD then
19:           (BestMP, BestUMP, BestNHD) ← (MP, UMP,NHD)
20:          end if
21:         end if
22:        end if
23:     end for
24:    end for
25:    BSM ← CompareBestPair(BSM, rs, ts, BestMP, BEstUMP,
          BestNHD)
26:   end for
27:   FMT ← FMT∪BSM
28: end for
```

Figure 4. Pseudo Code for SMTP

B. Grassfire Spot Matching Algorithm

STMP is very time consuming because every combination of two points respectively from reference gel and target gel is tested to find the right matched pairs. Besides, selecting the pivot pair similarly goes through trying every combination of two neighbor spots respectively from reference gel and target gel. Trying every combination is not only time-consuming but also prone to be stuck in false-positive results because there are many similar patterns which cause matches that are not exactly right.

The grassfire algorithm for spot matching is proposed to solve these kinds of problems [11][17]. The word "grassfire" means the matching process starts one single point and then spreads all around like wild fire does. It requires a seed spot pair as ground zero where a series of matching process starts. Matching process occurs with the exact information of the two center points and the pivot pair obtained from the previous step. This makes grassfire algorithm much faster and much more precise at the same time.

The grassfire algorithm starts from seed spot pair and neighbor spot pairs can be obtained. They go through spot matching processes to get spot matching results such as MP, UMP and NHD. Neighbor spot pairs with the matching results are stored into the matching information table. Neighbor spot pairs with poor matching results are chosen less because they are not thought to be the right match. Poor matching results are defined with thresholds for three parameters described above.

For the next step, the best neighbor spot pair is taken out from the information table. It is registered in the matching table which stores final matching results and it is used as the center spot pair. Spot matching is performed and neighbor

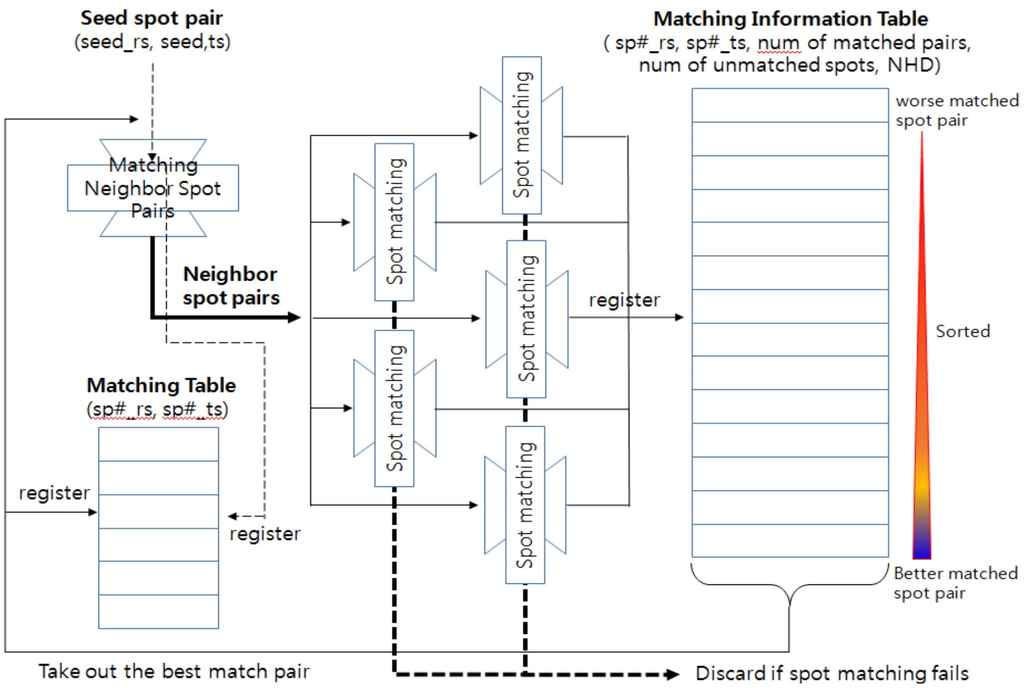


Figure 5. Conceptual Diagram of Grassfire Spot Matching Algorithm

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 9 (2017) pp. 1935-1942

spot pairs are obtained. For every neighbor spot pair, spot matching with neighbor spot pairs as the center spot pairs is performed and the pairs which meet the threshold are saved into matching information table as depicted in the figure 5. Matching results can be checked doubly with the help of matching table. If a matched neighbor pair is already exists in matching table or it conflicts with information in matching table, it should be discarded and does not stored into matching information table. The same process is repeated until the matching information table becomes empty. The grassfire algorithm is represented in the form of pseudo code in the figure 6.

```
m_Info_table = 0
m_pair_table = 0
m_pair_table←m_pair_table∪(r_seed, t_seed)
neighborpair←get_neighbor_pairs (r_seed, t_seed)
while (true) do
  for each spot pair (rs, ts)∈neighborpair do
    if (rs, ts) does not exist in m_Info_table then
      m_result←spot_matching (rs, ts)
      m_Info_table = m_Info_table∪(m_result, rs, ts)
    end if
  end for
  next_rs, next_ts←NULL
  m_resultmax←0
  for each (m_result, rs, ts) ∈ m_info_table do
    if m_result_max<m_result then
      if (rs, ts) does not exist in m_info_table then
        m_result_max←m_result
        (next_rs, next_ts) ← (rs, ts)
      end if
    end if
  end for
  if next_m_pair is NULL then
    break;
  end if
  m_pair_table←m_pair_table ∪ (next_rs, next_ts)
  neighborpair←get_neighbor_pairs (next_rs, next_ts)
end while
```

Figure 6. Pseudo Code for the Grassfire Matching Algorithm

PROPOSED METHOD

A. Things in Common between Spot Matching in 2-DE and Minutiae-based Fingerprint Matching.

In 2-DE, proteins are extracted as blobs. Blobs have a color, shape, size and center position parameters but only position is meaningful. So, it is frequently used because color, shape and size vary for the same protein. Central positions of blobs are extracted and gel images are transformed into a set of points. Then, point matching algorithms are applied to match two gels.

In minutiae-based fingerprint matching, minutiae are extracted from finger images. Minutiae have parameters such as direction, type and position. The three kinds of parameters are all meaningful but it can be though to be the exact same problem as spot matching in 2-DE if only position information is taken into consideration. STMP and the grassfire matching algorithm are made to consider this problem from the very beginning because local distortion in 2-DE could be the most difficult part. Finger print images are also known to contain local distortion which comes from the process of scanning of fingerprint even though it has less degree than that of 2-DE. The grassfire algorithm needs to start at a single seed spot pair and it is obtained manually or from the automatic pre-process [18]. When applying to the fingerprint matching, the core point in fingerprint images can be regarded as the seed spot pair.

B. Minutiae-based Fingerprint Matching using Grassfire Spot Matching

Image direction disparity of fingerprint is limited to less than 5 degrees in case of minutiae-based fingerprint matching just like in 2-DE. Image rotation is much less in 2-DE because gels are usually rectangles and dedicated scanners are used to obtain gel images. There are three parameters in minutiae but only position information is used as the grassfire matching algorithm does.

Matching conditions are set to be the same as in case of matching in 2-DE. It is because the grassfire algorithm is made originally for matching in 2-DE and it can be used without any modification. Matching starts from the core point as ground zero. In turn, neighbor points of the core points are matched in the matching process and they become center points in next step. The pair of core points is pivot pair for next step. Fingerprints have usually 20 to 30 minutiae and two fingerprints are considered to be matched if they have more than 10 matched minutiae.

EXPERIMENT AND RESULT

A. Fingerprint Preparation

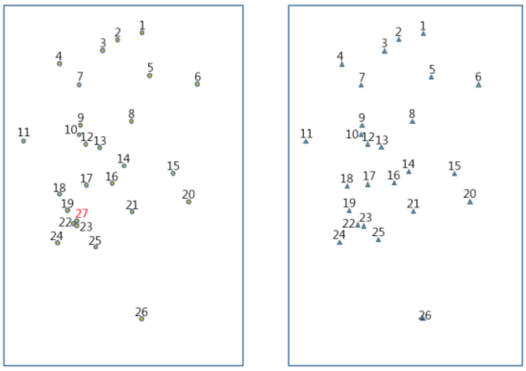
The experiment was performed only with one pair of fingerprint images from the same person as shown in the figure 7. It focused on verifying grassfire algorithm and showing possibilities that it can be used as fingerprint matching.

(a) Reference (b) Target

Figure 7. Fingerprint Images for Experiment

B. Minutiae Extraction

Minutiae are extracted manually for the two fingerprints. As depicted in the figure 8, the reference image of (a) contains 27 minutiae and the target image of (b) has 26 minutiae. Each pair is labeled with the same number to notice that two points are the corresponding one-to-one matching pair easily. Minutiae have only position information and other traits such as type and orientation are ignored because the grassfire algorithm is designed for using only position information of spots in 2-DE.



(a) Reference (b) Target

Figure 8. Minutiae Extraction

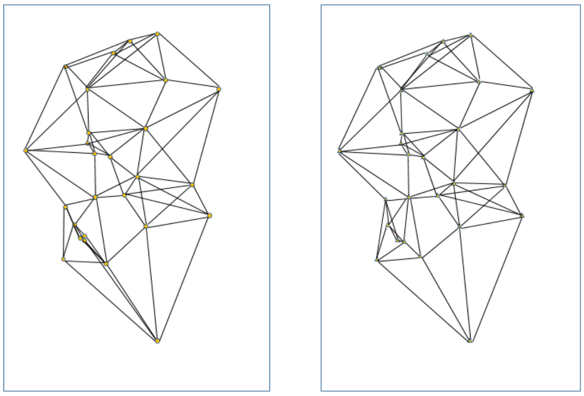
The table 1 shows x- and y-coordinates of minutiae in the figure 8. We can find out there are 26 matching pairs except 1 minutia. In the previous research, many minutiae extraction methods have been proposed together with minutiae matching algorithm. But extraction and matching are independent from each other and separate stages. That is why we extract minutiae in manual, not to present minutiae extraction algorithm. Core points should exist because they are used as the seed spot pair when applying to the grassfire algorithm. In this case, the point set of 23 from both reference and target minutiae becomes the core point.

Table 1. 2D Coordinates of Minutiae

#	Reference		Target		#	Reference		Target	
	x	*y*	*x*	*y*		*x*	*y*	*x*	*y*
1	209	41	203	42	15	257	254	251	254
2	172	52	166	51	16	164	268	160	269
3	150	68	145	69	17	126	273	119	271
4	85	88	81	88	18	85	286	89	274
5	221	107	215	108	19	97	311	92	311
6	293	118	287	120	20	280	298	274	296
7	115	120	109	120	21	194	313	189	313
8	194	175	186	175	22	105	330	104	332
9	117	181	111	182	23	111	334	114	334
10	114	195	109	195	24	82	359	77	359
11	30	205	25	205	25	139	336	136	354
12	124	211	119	210	26	209	475	203	474
13	145	215	140	214	27	111	326	-	-
14	182	243	181	251					

C. Building Graphs of 5-NNG

Two corresponding 5-NNG graphs are built from reference point set and target ones, respectively. The figure 9 shows the configuration of two graphs, from which information for the neighbor points can be obtained. Neighbor points are defined by connecting edges that are formed based on the graph theory of characterizing 5-NNG. In other words, one point forms edges with top nearest five points around it in 5-NNG. Sometimes, one point can have more than five edges because they are formed between two points and some edges are not in common.



(a) Reference (b) Target

Figure 9. Graph Configuration of 5-NNG

D. Execution of Grassfire Matching Algorithm

The grassfire matching algorithm is implemented using the programming language Perl. It is performed by reading two text files as input. They contain records as many as the number of points including x-coordinate, y-coordinate and point index. And point indexes are labeled with the same number in order to verity matching results easily.

The algorithm starts from seed spot pair(#23, #23). And then, matching process is spreading around the seed spot pair. The accuracy of spot matching is improved better by using matching information of previous stages. After completing the infinite iteration, matching results are printed out in the form of matched pair(reference_point_number, target_point_number). Finally, each of them should be examined to verify its correctness whether two minutiae have the same index or not.

E. Minutiae Matching Result

The table 2 shows the result of the experiment. There are 26 pairs both the reference and the target minutiae point set. In particular, an additional single point exists in the reference point set contrast to the target one.

Table 2. Minutiae Matching Result

Reference	Target	Reference	Target
1	1	14	15
2	2	15	15
3	3	16	16
4	4	17	17
5	5	18	18
6	6	19	19
7	7	20	20
8	8	21	21

9	9	23	23
10	10	24	24
11	11	25	25
12	12	26	26
13	13	27	22

The total of 26 point pairs are detected by the algorithm. Among them, 25 point pairs are true-positive matches. Whereas, only one pair (#27, #22) turns out to be a false-positive match which is located in the vicinity of the core point in the fingerprint. From the result, the algorithm works well showing dominant matching performance when applied to the minutiae-based fingerprint matching. Because it not easy for human to match the points in this area when there is only one single point which does not have the right match and points are densely populated.

CONCLUSION

In this paper, the grassfire algorithm is applied to match minutiae-based fingerprints. The grassfire algorithm is originally proposed for spot matching in 2-DE. It shows great possibility for fingerprint matching problems because there are many things in common between spot matching in 2-DE and fingerprint matching.

Only point matching is focused in this paper. So, we performed the experiment by applying the grassfire algorithm without considering other properties of minutiae. In this circumstance, it is obvious that the algorithm works well for minutiae-based fingerprint matching. As for the other properties of minutiae, types of minutiae are prone to be falsely detected depending on minutiae detecting algorithms or image quality, while direction information of minutiae is very robust and less error prone. Nevertheless, we will keep on trying to improve the grassfire algorithm to employ the other features for better performance in accuracy and speed.

ACKNOWLEDGEMENT

The corresponding author is Wook Hyun Kim.

REFERENCES

[1] B. Miller, "Vital signs of identity," IEEE Spectrum, Vol. 31, No. 2, pp. 22-30, February 1994

[2] Prabhakar, Salil, "Fingerprint classification and matching using a filterbank, " Diss. Michigan State University, 2001

[3] Ratha, Nalini K., Shaoyun Chen and Anil K. Jain. "Adaptive flow orientation-based feature extraction in fingerprint images," Pattern Recognition, Vol. 28, No. 11, pp. 1657-1672, 1995

[4] Lee, Henry C., Robert Ramotowski and R. E. Gaensslen, eds, "Advances in fingerprint technology," CRC Press, 2001

[5] N. Yager and A. Amin, "Fingerprint verification based on minutiae features: A review," Pattern Analysis and Applications, Vol. 7, Issue 1, pp. 94–113, 2004

[6] Popović, Brankica M. and Ljiljana Mašković. "Fingerprint minutiae filtering based on multiscale directional information," Facta universitatis-series: Electronics and Energetics, Vol. 20, No. 2, pp. 233-244, 2007

[7] http://syhrl.blogspot.kr/2012/04/036-fyp-minutiae.html

[8] Ismail, Ovais and Bhawani Singh Shekhwat. "Robust Fingerprint Matching," International Journal of Computer Applications, Vol. 1, No. 21, pp. 39-24, 2010

[9] Jianjiang Feng, Anil K. Jain and Karthik Nandakumar, "Fingerprint Matching," Computer, Vol. 43, pp. 36-44, February 2010

[10] YU, Kyung Deok, NA, Sangsin and CHOI, Tae Young, "A fingerprint matching algorithm based on radial structure and a structure-rewarding scoring strategy," International Conference on Audio-and Video-Based Biometric Person Authentication. Springer Berlin Heidelberg, pp. 656-664, 2005

[11] Liu, Ning, Yilong Yin and Hongwei Zhang. "A fingerprint matching algorithm based on Delaunay triangulation net," Computer and Information Technology, The Fifth International Conference, IEEE, 2005

[12] F. Chen, J. Zhou and C. Yang, "Reconstructing Orientation Field from Fingerprint Minutiae to Improve Minutiae-Matching Accuracy," IEEE Transactions on Image Processing, Vol. 18, No. 7, pp. 1665-1670, JULY 2009

[13] M. M. Min and Y. Thein, "Intelligent Fingerprint Recognition System by Using Geometry Approach," Proceedings of the IEEE, 2009

[14] F. Chen, J. Zhou and C. Yang, "Reconstructing Orientation Field From Fingerprint Minutiae to Improve Minutiae-Matching Accuracy," IEEE Transactions On Image Processing, Vol. 18, No. 7, pp. 1665-1670, JULY 2009

[15] J. Qi, M. Xie and W. Wang, "A Novel Fingerprint Matching Method Using a Curvature-Based Minutia Specifier," Proceedings of the 15th International IEEE Conference on Image Processing 2008, pp. 1488-1491, 2008

[16] Yun-Kyoo Ryoo, *et al.*, "Grassfire Spot Matching Algorithm in 2-DE," International Journal of Bio-Science & Bio-Technology, Vol. 5, No. 4, 2013

[17] Chan-Myeong Han, *et al.*, "A Spot Matching Algorithm using the Topology of Neighbor Spots in 2D-PAGE Images," International Journal of Software Engineering and Its Applications, Vol. 7, No. 5, 2013

[18] Dae-Seong Jeoune, *et al.* "Reliability Measure of 2D-PAGE Spot Matching using Multiple Graphs." Proceedings of the International Conference on Image Processing, Computer Vision and Pattern Recognition(IPCV). World Congress in Computer Science, Computer Engineering and Applied Computing (WORLDCOMP), 2014