

Infrared Face Vascular ID :

Vascular structure matching algorithm

Directed Studies Course Presentation
Supervisor: Professor Anwar Haque

Aasminpreet Singh Kainth

Department of Computer Science
Middlesex College, Western University

Agenda

Motivation

Advantages in IR

Challenges in IR

Previous work

Description of Problems (Use cases)

Research Question

Biometric Graph Matching Algorithm

BGR Algorithm Outline

BGC Algorithm Outline

Solutions

Future Work

Motivation



MOBILE ID NEWS SOLUTIONS APPLICATIONS EXCLUSIVE COMPANIES EVENTS

Search

Apple Patent Would Add Vein Biometrics to Face ID

July 21, 2020

Apple filed a patent this week that looks to improve upon facial recognition by mapping the unique vein patterns beneath a person's face.

The patent – titled “Vein matching for difficult biometric authentication cases” – calls for the use of an infrared sensor to capture sub-epidermal vein patterns beneath the user’s face in a 3D map, and then using the data from that map for authentication.

The iPhone’s current form of facial recognition, Face ID, does in fact use infrared sensors in its camera to map a user’s face, but focuses on the surface rather than beneath it.

Though Face ID and Apple’s fingerprint solution Touch ID – and most any form of biometric authentication – are a vast improvement on the older PIN/password approach, there have still been cases of hackers successfully fooling both systems through sometimes complex and



Premier Partners



Listen To The ID Talk Podcast



Privacy - Terms

Research

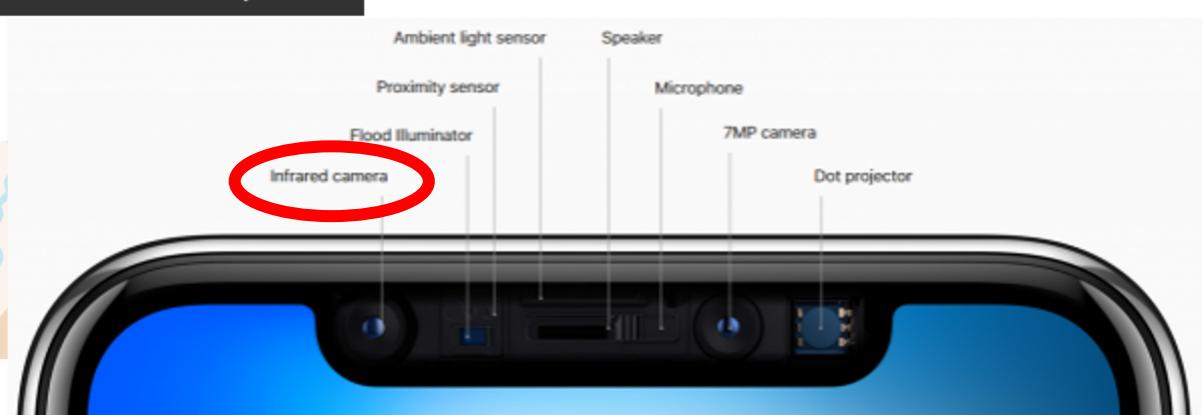


The screenshot shows the header of the MobileIDWorld website. It features a light blue header bar with the "MobileIDWorld" logo on the left, which includes a stylized network icon. To the right of the logo is a white search bar containing the word "Selphi" and the tagline "YOU BLINK YOU'RE IN" next to a blue eye icon. Below the header is a navigation menu with links: MOBILE ID, NEWS, SOLUTIONS, APPLICATIONS, EXCLUSIVE, COMPANIES, and EVENTS. To the right of the menu is a search bar with a magnifying glass icon.

The patent – titled “Vein matching for difficult biometric authentication cases” – calls for the use of an infrared sensor to capture sub-epidermal vein patterns beneath the user’s face in a 3D map, and then using the data from that map for authentication.

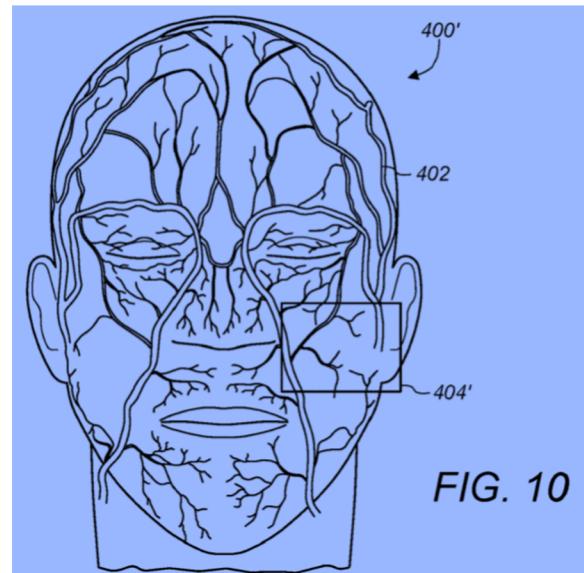
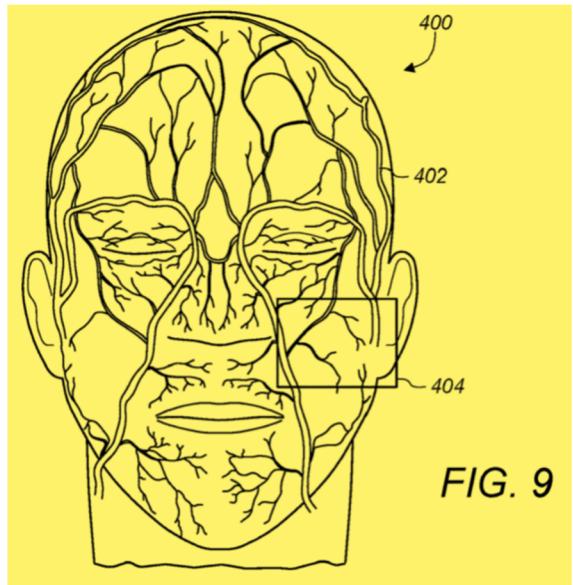
The iPhone’s current form of facial recognition, Face ID, does in fact use infrared sensors in its camera to map a user’s face, but focuses on the surface rather than beneath it.

Though Face ID and Apple’s fingerprint solution Touch ID – and most any form of biometric authentication – are a vast improvement on the older PIN/password approach, there have still been cases of hacking successfully fooling both systems through convincing冒充 and

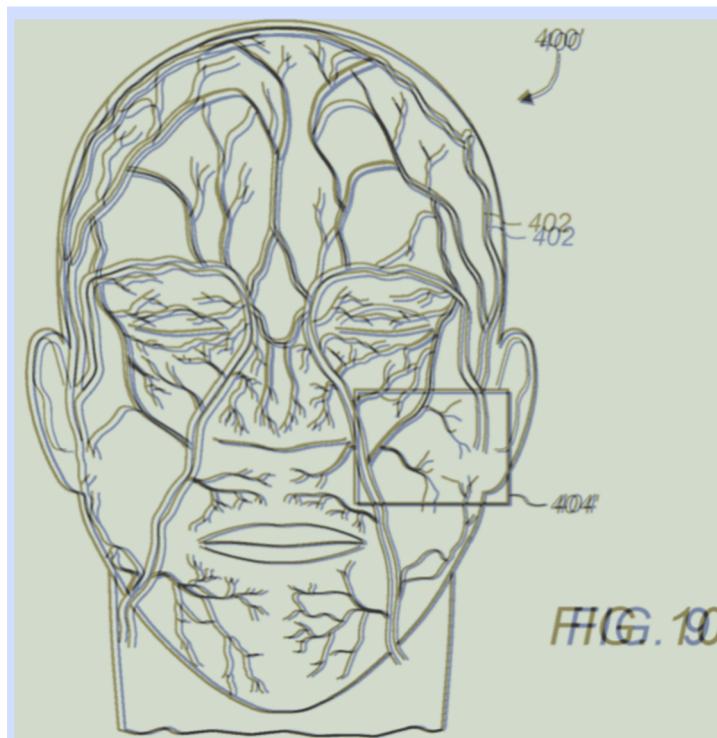


Apple X Front Camera Technology

Research



Research



Matching the Vascular graph

Advantages in IR

The foremost advantage of IR data in comparison with conventional visible spectrum images in the context of face recognition lies in its **invariance** to visible spectrum illumination. This is inherent in the very nature of IR imaging and, considering the challenge that variable illumination conditions present to face recognition systems, a major advantage.

IR energy is also **less affected by scattering** and absorption by smoke or dust than reflected visible light [1].

Unlike visible spectrum imaging, IR imaging can be used to extract not only exterior but also useful **subcutaneous**(situated-under-the-skin) anatomical information, such as the vascular network of a face [2].

Finally, thermal vision can be used to **detect facial disguises** [3] as well.

[1] H. Chang, A. Koschan, M. Abidi, S. G. Kong, and C. Won. Multispectral visible and infra-red imaging for face recognition; In Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 1–6, 2008.

[2] F. Nicolo and N. A. Schmid. A method for robust multispectral face recognition.; In Proc. IEEE International Conference on Image Analysis and Recognition (ICIAR), 2 :180–190, 2011.

[3] I. Pavlidis and P. Symosek. The imaging issue in an automatic face/disguise detection system. In Proc. IEEE Workshop on Computer Vision Beyond the Visible Spectrum (CVBVS), pages 15–24, 2000.

Challenges in IR

The use of IR images for automatic face recognition is not void of its problems and challenges:

MWIR and LWIR images are **sensitive to the environmental temperature**, as well as the emotional, physical and health condition of the subject.

Eyeglasses are opaque to the greater part of the IR spectrum (LWIR, MWIR and SWIR). This means that a large portion of the face wearing eyeglasses may be occluded, causing the loss of important discriminative information.

Impact of sunlight if recognition is performed outdoors and during daytime. Although invariant to the changes in the illumination by visible light itself, the IR “appearance” in the SWIR sub-bands is affected by sunlight which has significant spectral components at the corresponding wavelengths.



Fig. In infrared thermography, there are three wavelengths that can be broken down even further to specific wavelength ranges. The three wavelengths are: Short Wavelength Infrared (SWIR), Medium Wavelength Infrared (MWIR), and Long Wavelength Infrared (LWIR).

Previous Work

The earliest attempts at examining the potential of infrared imaging for face recognition date back to the work done by Prokoski et al. in 1992.

Most of the automatic methods which followed closely mirrored the methods developed for visible spectrum based recognition. Generally, these used holistic face appearance in a *simple statistical manner*, with little attempt to achieve any generalization.

The method of Wu et al. is one of the few in the literature which attempts to extract useful *subcutaneous* information from IR appearance. Using a series of assumptions on relative temperatures of body's deep and superficial tissues, and the ambient temperature, Wu et al. formulate a *differential equation* governing blood perfusion.

$$\omega = \frac{\varepsilon\sigma(T_s^4 - T_e^4) + A\mu d^{3M-1}(P g\beta/\nu^2)^M (T_s - T_e)^{M+1} - k(T_c - T_s)/D - H_m}{\alpha c_b(T_a - T_s)}$$

The model is then used to compute a "*blood perfusion image*" from the original segmented thermogram of a face.

Finally, blood perfusion images are matched using a standard linear discriminant.



(a) Thermal data ($T_e = 26.2^\circ\text{C}$);



(b) Corresponding blood perfusion data

Fig. Thermal data vs blood perfusion data (method of Wu et al.)

Previous Work

One of the most prominent methods which has explored the use of invariant anatomical features was developed by Buddharaju et al.

The **key observation** behind their method is that blood vessels are somewhat warmer than the surrounding tissues, allowing them to be identified in thermograms. These temperature differences are very small and virtually imperceptible to the naked eye, but inherently maintained regardless of the physiological state of the subject.

An important property of vascular networks which makes them particularly attractive for use in recognition is that the blood vessels are "**hardwired**" at birth and form a pattern which remains virtually unaffected by factors such as aging, except for predictable growth.

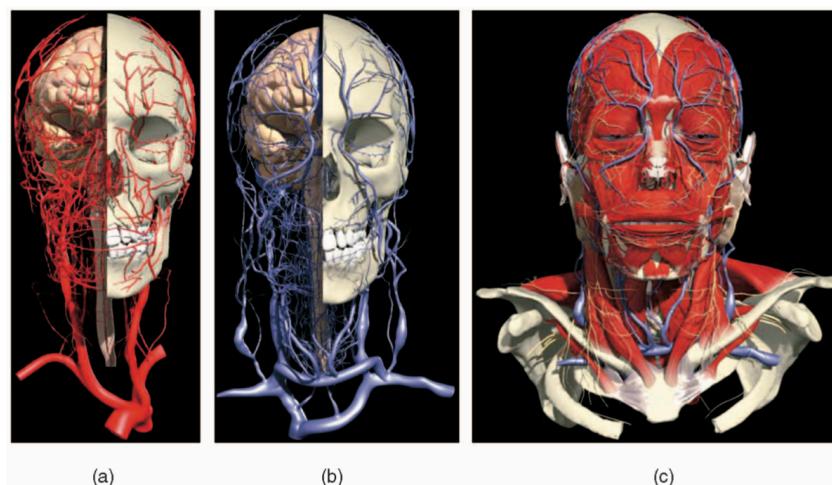


Fig.. Generic map of superficial blood vessels on the face (a) Overview of an arterial network. (b) Overview of a venous network. (c) Arteries and veins together underneath the surface of the facial skin.

Previous Work

Following automatic background-foreground *segmentation* of a face, Buddharaju et al. first extract blood vessels from an image using *simple morphological filters*. The **skeletonized vascular network** is then used to localize salient features of the network which they term *thermal minutia points* and which are similar in nature to the minutiae used in fingerprint recognition. Indeed, the authors adopt a method of matching sets of minutia points already widely used in fingerprint recognition, using relative minutiae orientations on local and global scale.

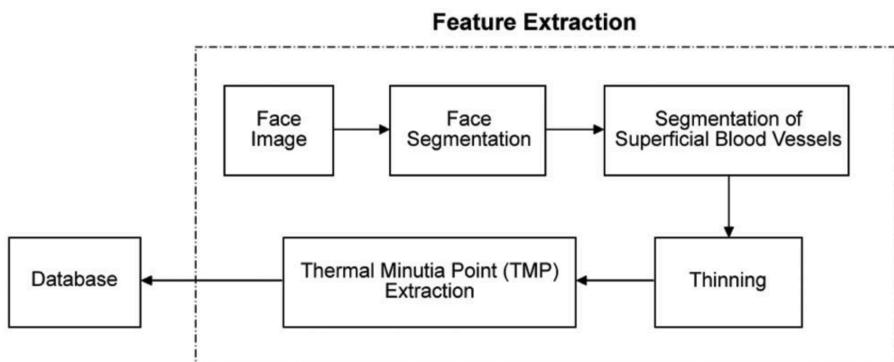


Fig. Architecture of feature extraction algorithm

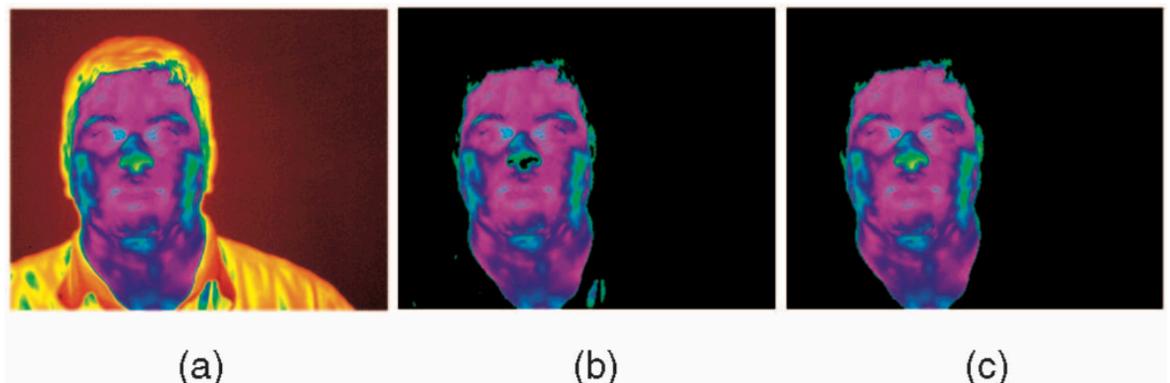


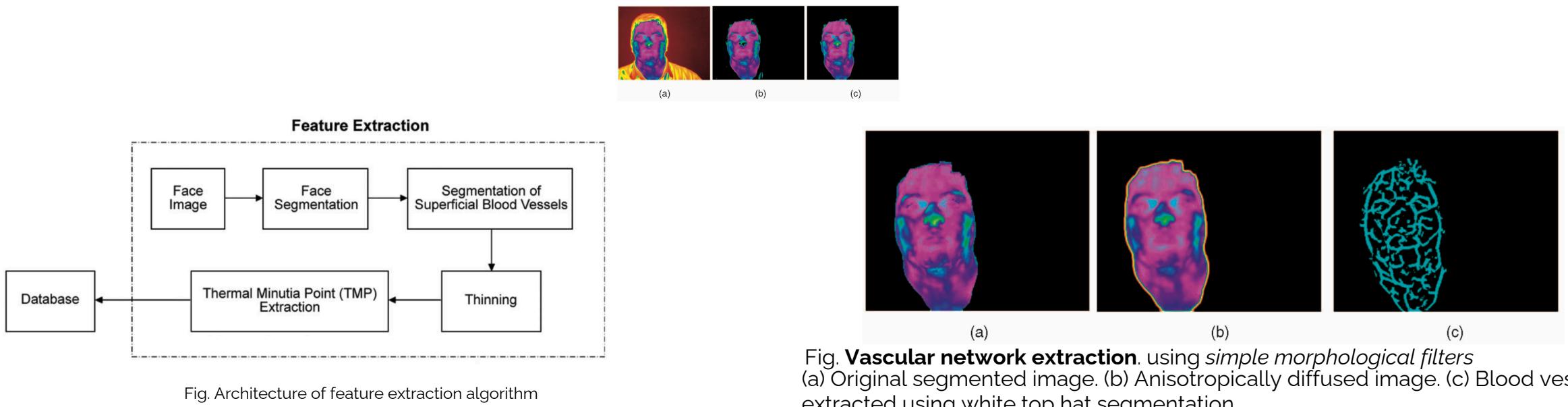
Fig. **Segmentation of facial skin region.**

(a) Original thermal facial image. (b) Result of Bayesian segmentation, where background is depicted in black. (c) Result of postprocessing.

Feature extraction algorithm

Previous Work

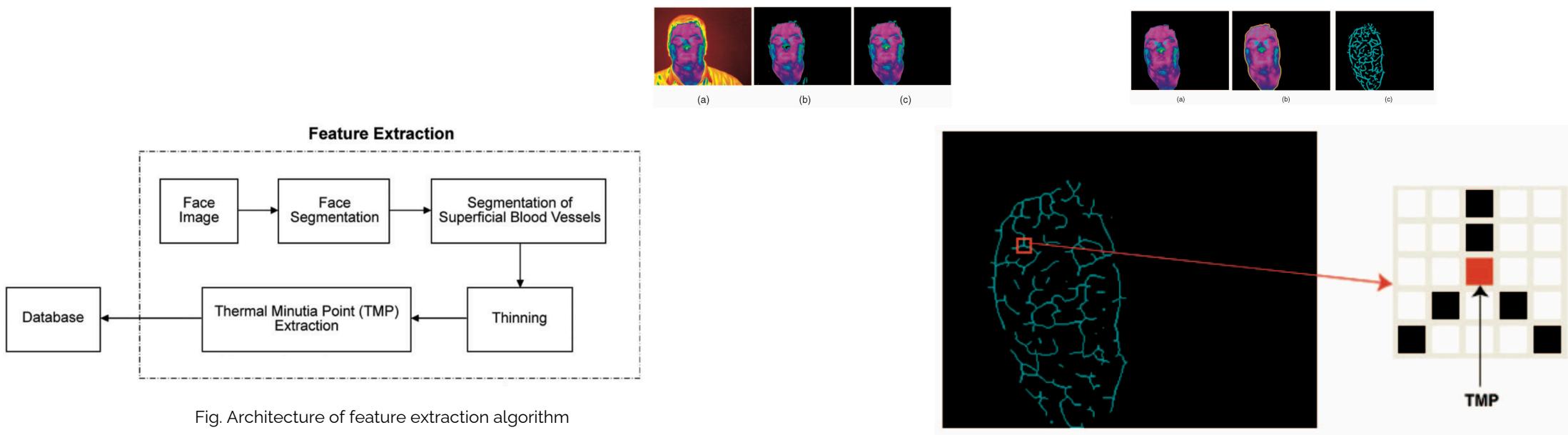
Following automatic background-foreground *segmentation* of a face, Buddharaju et al. first extract blood vessels from an image using *simple morphological filters*. The **skeletonized vascular network** is then used to localize salient features of the network which they term *thermal minutia points* and which are similar in nature to the minutiae used in fingerprint recognition. Indeed, the authors adopt a method of matching sets of minutia points already widely used in fingerprint recognition, using relative minutiae orientations on local and global scale.



Feature extraction algorithm

Previous Work

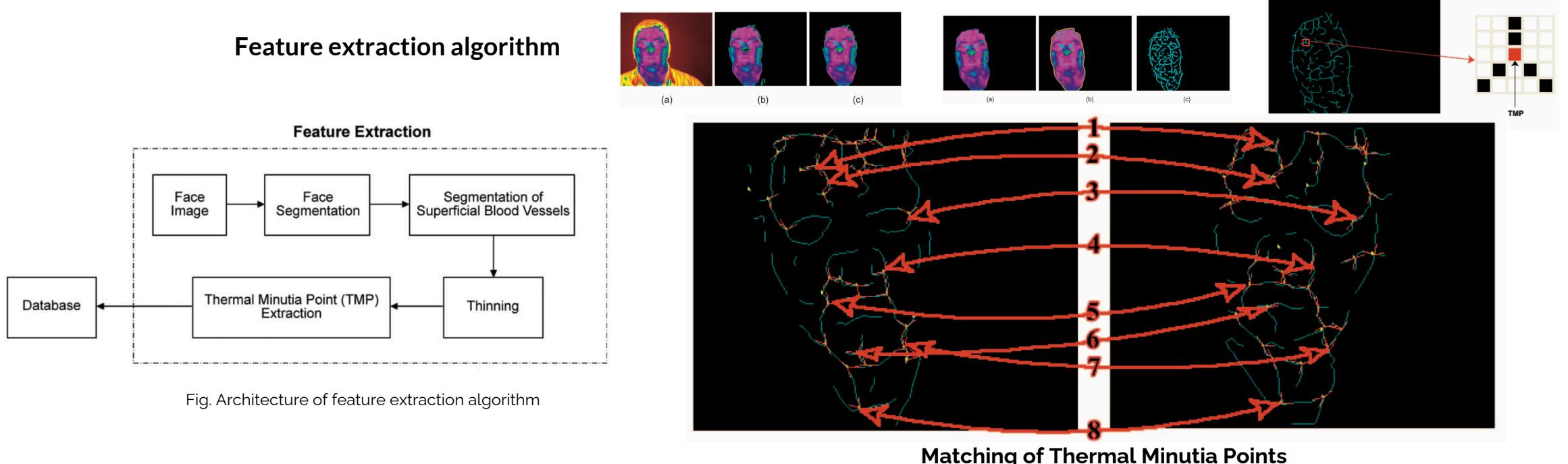
Following automatic background-foreground segmentation of a face, Buddharaju et al. first extract blood vessels from an image using *simple morphological filters*. The **skeletonized vascular network** is then used to localize salient features of the network which they term *thermal minutia points* and which are similar in nature to the minutiae used in fingerprint recognition. Indeed, the authors adopt a method of matching sets of minutia points already widely used in fingerprint recognition, using relative minutiae orientations on local and global scale.



Feature extraction algorithm

Previous Work

Following automatic background-foreground segmentation of a face, Buddharaju et al. first extract blood vessels from an image using *simple morphological filters*. The **skeletonized vascular network** is then used to localize salient features of the network which they term *thermal minutia points* and which are similar in nature to the minutiae used in fingerprint recognition. Indeed, the authors adopt a method of matching sets of minutia points already widely used in fingerprint recognition, using relative minutiae orientations on local and global scale.



Matching is a two-step process

1. vascular network of the test is registered with the database subjects using the *dual bootstrap ICP Algorithm*
2. to generate matching score. Score is generated by **pixel to pixel comparison** of the test and database vascular networks.

Description of Problem

1. The matching algorithm that compares the registered vascular network of test and database subjects is based on a **simple heuristic pixel to pixel comparison**.
2. The output produced by Buddharaju's approach is very sensitive to the scale of the face and thus **lacks robustness** to the distance of the user from the camera.
3. Buddharaju et al. produced a binary image, in which each pixel is deemed either as belonging to the vascular network or not, *without any accounting for the uncertainty* associated with this classification. This aspect of their approach makes it additionally **sensitive to variable pose and temperature changes** across the face, and time-lapse.
4. They did not test the **performance of their system after time-lapse**. Their results were reported just for the same session data. As they did not consider any uncertainty for the output of the pixels in vascular network, some changes can occur in their binary vascular network after time-lapse. This can have a dramatic effect in the performance of their system.
5. They did not propose any model to **normalize the changes caused by facial expression and pose**. For the case of pose, they trained their algorithm using multiple images, and the pose of an input face was first categorized by pose. Next, it was matched with training faces in that pose only (they adopted their feature matching algorithm from fingerprint recognition which are fixed in nature).
6. The method by which they solved the **eyeglasses problem** is questionable. The probe images wearing glasses were compared to their corresponding thermal images with glasses in the gallery, while the subjects in the gallery are preferred to not to wear eyeglasses due to the occlusion caused by eyeglasses.

Research Question

1. How can we extract an anatomy based identity descriptor (vascular network) which is robust to distance (scale) changes and time-lapse, and accounts uncertainty to the pixels for belonging to the vascular network (i.e., a gray level vascular network rather than a binary one)?
2. Is it possible to *normalize the facial expression and pose* in thermal faces using some sophisticated models? Thermal IR images of faces are much less rich in fine detail than visible spectrum images. This makes the problem of model fitting all the more challenging.

Biometric Graph Extraction

The basic features in a vascular graph are *vein terminations* (where the veins go out of the frame of reference), *vein bifurcations* (where one vein splits into two) or *vein crossovers* (where two veins appear to intersect in the skeleton).

A biometric graph is a spatial graph with the vein features of terminations, bifurcations and crossovers forming the graph vertices. A pair of vertices will have an edge between them if we can trace along a vein from one vertex to another, without encountering any other vertex in between. Fig. shows examples of biometric graphs from the four modalities.

To construct the Biometric Graph from a two-dimensional biometric image, the vessel skeleton is extracted from the image and the feature points are found. The feature points are labelled to form the vertex set, and their coordinates are recorded. The existence of an edge between vertices is determined by tracing the skeleton from each feature point until another is encountered. The length and slope of each edge is calculated and recorded. Other feature point and vessel segment attributes can be calculated at the same time.

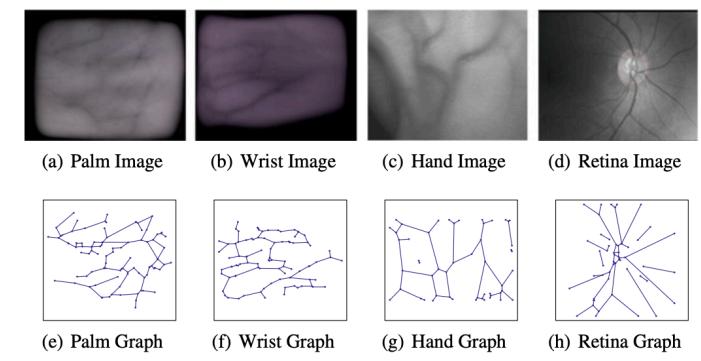


Fig. Vascular patterns from four modalities **a** Palm **b** Wrist **c** Hand and **d** Retina vessels and their corresponding spatial graphs in **(e-h)**

Biometric Graph Matching Algorithm

The algorithm has two parts: **BGR (Registration)** which requires 4 steps; and **BGC (Comparison)**, in which the 3 steps are finding the graph edit distance, identifying the Maximum Common Subgraph (MCS) and scoring comparisons using graph-based difference measures.

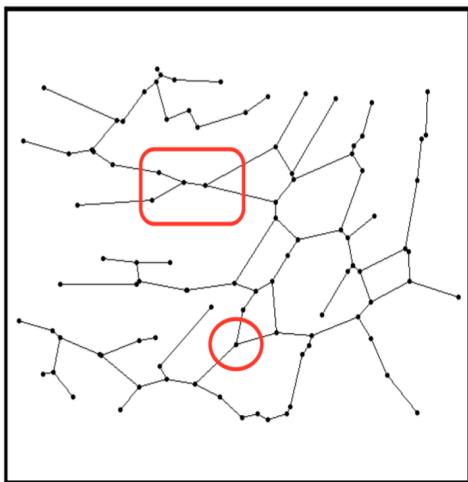


Fig. Example of graphs extracted from the vascular biometric modality.

The red circle in the palm vein graph shows an example of a claw structure. The red square shows an example of a two-claw structure. A two-claw structure is formed by a pair of star structures sharing a common edge

BGR Algorithm Outline

The four design parameters in the BGR algorithm are a *structure S*, a *similarity score function f* depending on the structure selected, a *structure pair shortlist length L* and a *vertex comparison tolerance ε* .

The structures S used are: Edges (E), Claws (C) and Two-claws (T).

BGR (S, f, L, ε) is used to denote the algorithm

Step 1: Initialisation Select S, f, L and ε . The two graphs g and g' to be registered are inputs to the algorithm. The registration process begins by identifying and listing all the structures of the selected type S in each graph.

Step 2: Similarity scoring structure pairs Each structure in the first graph g and structure in the second graph g' is compared using F to obtain a similarity score. The similarity function chosen depends on the structure.

Require: Graphs g and g' with vertex sets $\mathbf{V} = \{v_1, v_2, \dots, v_m\}$ and $\mathbf{V}' = \{v'_1, v'_2, \dots, v'_{m'}\}$ and vertex sets $\mathbf{E} = \{e_1, e_2, \dots, e_n\}$ and $\mathbf{E}' = \{e'_1, e'_2, \dots, e'_{n'}\}$, respectively. Let L be the number of structure pairs to shortlist and let ε be the tolerance to match vertex pairs.

Ensure: Aligned graphs g_a and g'_a having same edge links as g and g' but with new spatial coordinates.

- 1: $g_a \leftarrow \emptyset$ and $g'_a \leftarrow \emptyset$. \triangleright Initialise the registered graphs that will be returned at the end of the algorithm
- 2: $\mathbf{S} = \{s_1, s_2, \dots, s_n\}$ is the list of structures in g .
- 3: $\mathbf{S}' = \{s'_1, s'_2, \dots, s'_{n'}\}$ is the list of structures in g' .
- 4: $M_{dist} \leftarrow 0$ \triangleright Initialise a matrix of size $n \times n'$ with zeros.
- 5: **for** $a = 1$ to n **do**
- 6: **for** $b = 1$ to n' **do**
- 7: $d_{ab} = \text{STRUCTDIST}(s_a, s_b, F)$ \triangleright This function returns the distance between the two structures. The flag F indicates if the structure is an edge, claw or two-claw.
- 8: $M_{dist}[a, b] \leftarrow d_{ab}$
- 9: **end for**
- 10: **end for**

BGR Algorithm Outline

Step 2: Similarity scoring structure pairs Each structure in the first graph g and structure in the second graph g' is compared using F to obtain a similarity score. The similarity function chosen depends on the structure.

For example,

when “**edge**” pairs are compared they are scored based on the similarity of their lengths only (if no pre-alignment is assumed) or of their lengths and slopes (if some pre-alignment is assumed).

When “**claw**” pairs are compared they are scored based on the similarity of the lengths of their three edges and two included angles.

When “**two-claw**” pairs are compared, the similarity of the corresponding claw structures and connecting edges determines the score.

- 2: $S = \{ s_1, s_2, \dots, s_n \}$ is the list of structures in g .
- 3: $S' = \{ s'_1, s'_2, \dots, s'_{n'} \}$ is the list of structures in g' .
- 4: $M_{dist} \leftarrow 0$ ▷ Initialise a matrix of size $n \times n'$ with zeros.
- 5: **for** $a = 1$ to n **do**
- 6: **for** $b = 1$ to n' **do**
- 7: $d_{ab} = \text{STRUCTDIST}(s_a, s_b, F)$ ▷ This function returns the distance between the two structures. The flag F indicates if the structure is an edge, claw or two-claw.
- 8: $M_{dist}[a, b] \leftarrow d_{ab}$
- 9: **end for**
- 10: **end for**

```

27: function STRUCTDIST( $s_a, s_b, F$ )  $d_{structPair} \leftarrow \emptyset$ 
28:   if  $F == \text{"edge"}$  then ▷ The length and slope of the edge
29:      $E_a \leftarrow (l_a, \theta_a)$ 
30:      $E_b \leftarrow (l'_b, \theta'_b)$ 
31:      $d_{structPair} = \frac{1}{0.5(l_a+l'_b)} \text{EUCDIST}(E_a, E_b)$  ▷ The Euclidean distance
            between the lengths  $l$  and  $l'$  and slopes  $\theta$  and  $\theta'$  of the vertex pair.
32:   end if
33:   if  $F == \text{"claw"}$  then ▷ The three edges of the claw in decreasing order of edge length
34:      $L_a \leftarrow (l_{1a}, l_{2a}, l_{3a})$ 
35:      $\Theta_a \leftarrow (\theta_{12a}, \theta_{23a})$  ▷ The angles between first and second vertex and the second and third vertex.
36:      $L'_b \leftarrow (l'_{1b}, l'_{2b}, l'_{3b})$ 
37:      $\Theta_b \leftarrow (\theta'_{12b}, \theta'_{23b})$ 
38:      $l_\delta \leftarrow \text{EUCDIST}(L_a, L'_b)$ 
39:      $a_\delta \leftarrow \text{EUCDIST}(\Theta_a, \Theta'_b)$ 
40:      $d = l_\delta + a_\delta$ 
41:      $d_{structPair} = d$ 
42:   end if
43:   if  $F == \text{"two-claw"}$  then ▷ A two-claw has two-claw structures connected by a common edge
44:      $L_a \leftarrow (l_{1a}, l_{2a}, l_{3a}, l_{4a}, l_{5a}, l_{6a})$ 
            ▷  $l_1$  and  $l_4$  are the longest edges of the first and second claw structures. The other two edges follow the longest edge in decreasing order of length.
45:      $\Theta_a \leftarrow (\theta_{12a}, \theta_{23a}, \theta_{45a}, \theta_{56a})$  ▷ The four internal angles, two each from each of the two-claws.
46:      $l_{*a}$  is the length of the connecting edge between the two-claws in structure  $a$  where  $* \in \{1, 2, 3, 4, 5, 6\}$ .
47:      $L'_b \leftarrow (l'_{1b}, l'_{2b}, l'_{3b}, l'_{4b}, l'_{5b}, l'_{6b})$ 
48:      $\Theta'_b \leftarrow (\theta'_{12b}, \theta'_{23b}, \theta'_{45b}, \theta'_{56b})$ 
49:      $l'_{*b}$  is the length of the connecting edge between the two-claws in structure  $b$  where  $* \in \{1, 2, 3, 4, 5, 6\}$ .
50:      $d_1 = \text{EUCDIST}(L_a[1 : 3], L'_b[1 : 3]) + \text{EUCDIST}(\Theta_a[1 : 3], \Theta'_b[1 : 3])$ 
51:      $d_2 = \text{EUCDIST}(L_a[4 : 6], L'_b[4 : 6]) + \text{EUCDIST}(\Theta_a[4 : 6], \Theta'_b[4 : 6])$ 
52:      $d_3 = \text{EUCDIST}(l_{*a}, l'_{*b})$ .
53:      $d_{structPair} = d_1 + d_2 + d_3$ 
54:   end if
            return  $d_{structPair}$ 
55: end function

```

BGR Algorithm Outline

Step 3: Shortlisting structure pairs and aligning on them

The structure pairs are ordered based on decreasing order of similarity score. The top L high scoring structure pairs (for $S = C$ or $S = T$) or the top $L/2$ short and top $L/2$ long edges (for $S = E$) are shortlisted for further processing.

For every shortlisted structure pair, the two graphs are *translated and rotated* so that a specific part of the structure becomes the origin of the reference frame.

For example,

if edges are used, the vertex with smaller x coordinate becomes the centre of the coordinate system and the other vertex defines the direction of the positive x-axis.

If claws are used, the centre of the claw becomes the origin while the longest edge defines the direction of the positive x-axis.

If two-claws are used, the connecting edge defines the coordinate system, again taking the vertex with smaller x coordinate as the origin of the reference frame.

```
11: Sort the contents of  $M_{dist}$  in increasing order.  
12:  $M_{shortlist}$  is a matrix with 3 columns.  
    Every row  $m_i$  stores the 3-tuple  $(d_{abi}, a_i, b_i)$ .  
     $d_{abi}$  is taken from the sorted  $M_{dist}$  with the first row of  $M_{shortlist}$ ,  $m_1$  having  $d_{ab1}$ ,  
    the smallest distance.  
     $a_i$  and  $b_i$  indicate the corresponding row and column of  $d_{abi}$  in  $M_{dist}$ .  
13:  $d_{struct} \leftarrow (0, 0, \dots, 0)_{1 \times L}$      $\triangleright$  A vector to store the distances between graphs  
    when aligned on each of the shortlisted structure pairs  
14: for  $i = 1$  to  $L$  do  
15:    $a = a_i, b = b_i$  where  $m_i \in M_{shortlist}$   
16:    $g_o = \text{TRANSROT}(g, e_a)$ .           $\triangleright$  Translate and rotate  $g$  with respect to the  
    specific edge in the shortlisted structure  
17:    $g'_o = \text{TRANSROT}(g', e'_b)$ .  
18:    $d_{struct}[i] = \text{QUICKSCORE}(g_o, g'_o, \varepsilon)$   $\triangleright$  Compute a distance based on vertex  
    correspondence between the translated and rotated graphs  
19: end for  
  
56: function  $\text{TRANSROT}(g, e)$   
57:    $g_o \leftarrow g$   
58:   The vertex of  $e$  with the smaller  $x$  coordinate will be the origin of the coor-  
     dinate system.  
59:   The edge  $e$  will be define the positive direction of the x-axis.  
60:   Recalculate all the vertex attributes of  $g_o$  in the new coordinate system.  
   return  $g_o$ .  
61: end function
```

BGR Algorithm Outline

Step 4: Pair alignment scoring and graph registration

With both graphs in the same coordinate system, aligned on a shortlisted pair, each vertex in the first graph g is matched to a vertex in the second graph g' by finding the first vertex in g' that is within ε pixels from it.

If a vertex in g does not find a corresponding vertex in g' within ε pixels of it, it will not be matched.

The total number of matched vertices is normalized by the geometric mean of the number of vertices in the two graphs to provide a rough measure of alignment we call QuickScore (QS).

That is, if g has n vertices, g' has n' vertices and the aligned graphs have c matched vertices within tolerance ε , the distance between g and g' is calculated to be

$$QS(g, g') = 1 - \frac{c}{\sqrt{n \times n'}}$$

The pair of structures that gives the smallest score is chosen to register g and g' . The resulting registered graphs are denoted g_a and g'_a .

```
20:  $d_{min} = \text{MIN}(d_{struct})$ .
21:  $a_{min}$  and  $b_{min}$  are the row and column in  $M_{shortlist}$  corresponding to  $d_{min}$ .
22:  $g_a = \text{TRANSROT}(g, e_{a_{min}})$ .
23:  $g'_a = \text{TRANSROT}(g', e'_{b_{min}})$ .
      return  $g_a, g'_a$  and  $d_{min}$ .
```



```
62: function QUICKSCORE( $g, g', \varepsilon$ )
63:   Label all vertices of  $g$  and  $g'$  as unmatched.
64:    $C = 0$             $\triangleright$  Counter for number of vertex pair matches between  $g$  and  $g'$ 
65:   for  $i = 1$  to  $m$  do
66:     for  $j = 1$  to  $m'$  do
67:       if  $v_i$  is labelled unmatched and  $v'_j$  is labelled unmatched and EU-
        CDIST( $q_i, q'_j$ )  $\leq \varepsilon$  then
68:          $C = C + 1$ .                                $\triangleright v_i$  matches with  $v'_j$ .
69:         Label  $v_i$  and  $v'_j$  as matched.           $\triangleright q_i = (q_{1i}, q_{2i})$  is the vertex
        attribute of  $v_i$  and  $q'_i$  is the vertex attribute of  $v'_i$  .
70:       end if
71:     end for
72:   end for
73:    $d = 1 - \frac{C}{\sqrt{m \times m'}}$ . return  $d$ .
74: end function
```

BGR Algorithm Result

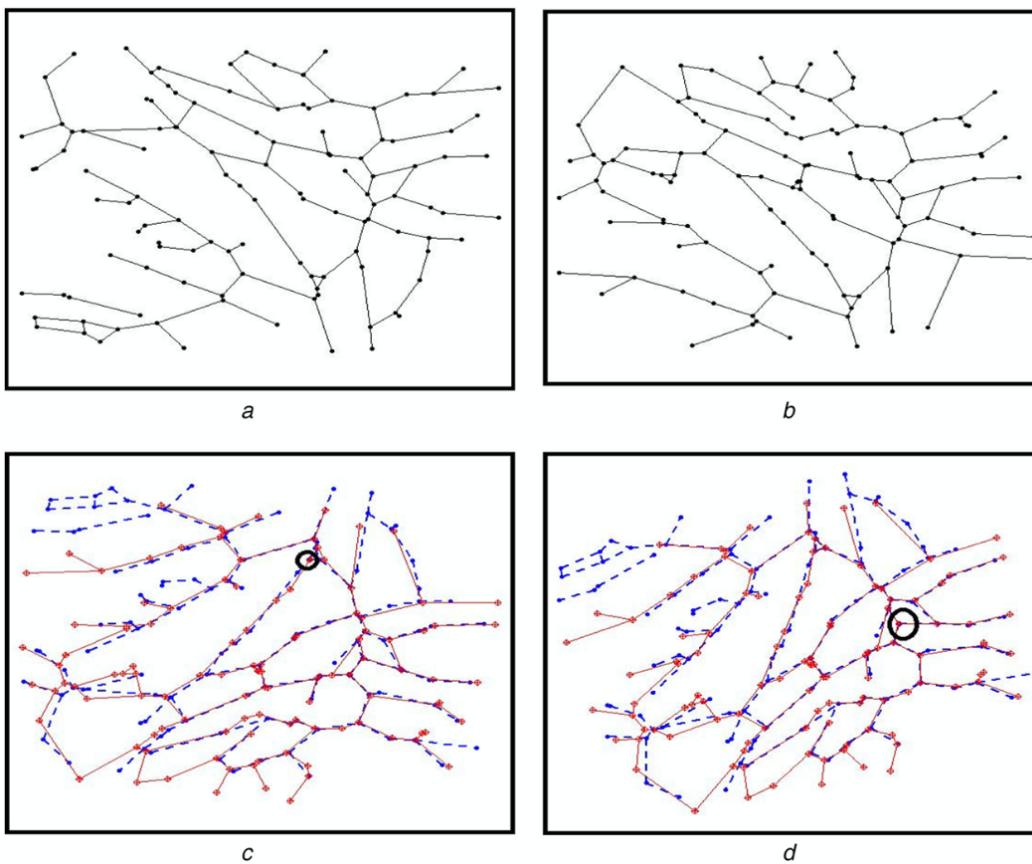


Fig. Registration process for a pair of palm vein graphs in (a) and (b). The bottom figures show registration using (c) the two-claw structure and (d) the claw structure.

The structure pair on which the graphs are aligned are shown with a black circle. In this particular example, registration with the two-claw structure gave a larger number of matched vertex pairs than the claw structure.

BGC-Biometric Graph Comparison

After the graphs are registered, the pair of noisy graphs are compared using an inexact graph matching technique that computes the graph edit distance between the pair. To do this the *Hungarian algorithm* based method proposed by Riesen and Bunke can be used.

It has three steps: determination of the minimum graph edit path between g_a and g'_a , construction of the Maximum Common Subgraph (MCS) of g_a and g'_a , and finally, measurement of the difference between g_a and g'_a using the MCS.

One graph can be converted to another by three operations – insertions of features, deletions of features and substitutions of features and have associated costs.

The selection of the right costs for these operations is critical to getting a meaningful measure of distance. In order to define the costs for all substitutions, insertions and deletions, Riesen and Bunke introduced the cost matrix.

The cost matrix is then fed into an *optimisation algorithm* and the output will be a list of edit operations that give the lowest cost.

BGR Algorithm Result

The matching algorithm that is proposed can provides better recognition accuracy compared to point pattern approaches from the previous works.

In my opinion, Graph registration is the key component of the algorithm, and is more critical than the graph comparison component. Although it can often be assumed that the capture mechanism enforces an approximate alignment of biometric images in the first place, literature tells that alignment is seldom ideal, and large differences can occur between captures from the same person, particularly as the time between captures increases. Unless two extracted BGs from the same biometric instance can be aligned well, comparison cannot be effective. Essentially this is because the need of a good similarity score for a genuine match, in order to minimise the number of false non-matches.

As the output produced by Buddharaju's approach is very sensitive to the scale of the face and thus **lacks robustness** to the distance of the user from the camera but with the Biometric registration strategy proposed it can be solved.

As Buddharaju et al. results are **sensitive to variable pose**, the BGM can solve this problem too.

The important outcomes from the work are

1. that using graph structure in the registration algorithm can increase the speed and accuracy of registration;
2. that graph structure in the MCS can be exploited to increase recognition accuracy; and
3. that using multiple graph structures can improve similarity scores over single structures.

Future Work

The followings are suggested as future work:

1. Examining the scalability of the system on larger and challenging databases which simulate more realistic conditions.
2. Testing the time robustness of the proposed system on a larger number of subjects after a long period of time.
3. Comparing the results of our vascular network with that of radiological images in order to determine those parts of the vascular network which correspond to the real vascular network.

Questions?

Thank You.