M. Tech. (Computer Science) Dissertation Series

PROCESSING OF FINGER PRINT IMAGE

a dissertation submitted in partial fulfilment of the requirement for the M. Tech. (Computer Science) degree of the Indian Statistical Institute

By

DEVENDRA SINGH

under the supervision of

Mr. M. K. KUNDU

CONTENTS

ABSTRACT

A new method for preprocessing fingerprint images is proposed here . It uses a new Enhancement technique , which is suitable for any Image that contains oriented patterns . Enhancement step is followed by Ridge Crispening . The resultant Image contains a number of ridge breaks . In the next step spurious breaks are removed . The decisive conditions for removing these breaks are based on (1) Eight neighborhood information as well as on (2) directionality at the candidate point . Ridges are then Smoothed and Binarized in subsequent steps . Finally the two tone Thinning is performed to get the perfect demarcated Ridges and valleys . The method proposed is compared with some standard methods .

1 INTRODUCTION

Automatic fingerprint identification system comprises of two main steps namely (i) Processing of raw Image . and (ii) Classification or actual Identification . Fingerprints are considered as basic tool for positive identification of individuals . It has got use not only in law enforcement , but also in different other civilian and defense purposes . The significance of fingerprints in identification lies in skin structure , composed of ridges and valleys . The objective of preprocessing is to get a two tone Image which clearly demarcates ridges from valleys and at the same time preserves all information contained in the input Image which are necessary for the extraction of complete feature set . The objective of Identification system is to correctly identify a fingerprint irrespective of the orientation of Input Image . Such an Identification system would be based on the basic structural information and different characteristic features present in the Image rather than on a pixel by pixel comparison . Hence in preprocessing step we need to preserve only those informations which concerns with the basic type or the characteristic features of the fingerprint .

The block diagram for a typical Automatic Fingerprint identification system is given fig. 1 .

For comparison we have taken two standard methods which are well recognized . These are O' Gorman's method [4] & Mahtre's method [3] . First one is brought out by AT & T and is developed by O' Gorman and Nickerson . Result obtained by this method is quite good , but is complex and expensive . The method developed by Mahtre is very simple , but it requires arbitrary selection of different parameters . So far as the performance of this method is concerned , much depends on the values of these parameters . Selection of optimal values for these parameters in case of low quality Image is a very tough job . Hence performance of this method is very much sensitive of the Image quality .

The motivation behind our work is explained below . If we do binarization (segmentation) of the initial gray tone Image then in most of the cases we might end up with erroneous results . So we can think of a method where segmentation is interleaved with Smoothing & Thinning of ridges as well as repairing of spurious breaks incurred so far . Thus if we can do crispening of the Ridges and connection of Ridge breaks in the gray domain itself then apriori requirement for binarization can be avoided . With these ideas in mind we are proposing a new technique for extracting all necessary informations from the initial raw Image .

2 FINGER PRINT CHARACTERISTICS

Processing of Fingerprint Images is done with consideration of the main characteristics of Fingerprints . In this section the general characteristics of Fingerprint are described . Fingerprint consists of some what parallel ridges which vary slowly over the Finger . Ridge structure remains unchanged during ones lifetime .

A Fingerprint consists of three areas , namely , Core area , marginal area and base area . The ridges from these three areas meet at a triangular formation called Delta region . In the Delta region lies a point called Delta point , whereas the point lying at the center of core region is called core point . Another parameter defined is the ridge-count , which is the number of ridges crossing an imaginary line segment obtained by joining core point and delta point.

The two basic parameters used in Fingerprint classification are (1) the ridge flow in core region and (2) the number of delta points present in the Fingerprint . On the basis of these parameters fingerprints are classified into the following classes :

- (a) Plain arch .
- (b) Tented arch .
- (c) Loops .
  - (i) ulnar loops (pointing towards the thumb) .
  - (ii) radial loops (pointing away from the thumb).
- (d) Whorls .
- (e) Central pocket .
- (f) Elliptical whorls .
- (g) Double loop .
- (h) Accidentals .

Some typical examples of the different types of fingerprints are shown in fig. 3 .

Some local features such as ridge ends , forks , islands etc. are also used in the comparison of fingerprints . These local features are irregularities , which are called minutiae and they differ from print to print . They also remain unchanged during ones lifetime . Minutiae other than ridgeends & bifurcations are suppressed in many systems .

3 DESCRIPTION OF AVAILABLE METHODS

In this section O' Gorman's method and Mahtre's method are briefly described .

O' Gorman & Nickerson method [4]
This method make use of prominent direction of ridges around each candidate pixel . In fact it considered ten directions .

The method consists of following five steps :
Step 1 : In this step some Image parameters are specified by the user . Now filter masks in different directions along with some other parameters are computed by using these specified values . There are ten prominent directions and hence there are ten different enhancement masks . Each filter mask is chosen to be a square with the sidelengths equal to the period of the signal (here a ridge and the adjacent valley constitute the full wave) or the period plus 1 , whichever is odd . For the filter mask in the horizontal direction , the middle rows consists of coefficients to amplify ridges . The rows on both sides of the middle strips are chosen to negatively amplify the valleys . Apart from middle and side strips there might also be transition strips . Refer [4] for details regarding how these coefficients are determined .

step 2 : In this step local ridge orientation in the vicinity of each pixel position is found out . This done by taking the absolute sum of successive gray level differences for each of the directions over a window centered around the candidate pixel . For example the sum for horizontal direction is given as :

$$Sum0 = \sum_{\ell=j-4}^{j+4} \sum_{m=i-4}^{i+3} (a_{\ell,m} - a_{\ell+1,m})$$

From the sums for different directions , the slope of ridge in the vicinity of the point is found out . The value thus calculated lies between 0 and π .

Step 3 : This oriented Image contains large areas of constant orientation as well as high curvature regions where orientation is constant only for small area . Hence smoothing is done from low to high resolution . Local orientation consistency is measured within a window of size $w \times w$, where W is varied from $W = W_{max}/2$ to $w = 3$. The window sidelengths are halved at each level from low to high resolution . The window with side length W corresponds to max the maximum sized consistent region . This value is chosen from observation . Ofcourse the choice of W_{max} is critical .

Step 4 : Pixel by pixel enhancement filtering of the oriented image based on local orientation is done and this is followed by adaptive binarization .

Step 5 : This is the post-processing step . Here background noise is reduced by convolving the entire image with mask of same size as the enhancement or orientation determination mask .

In this method user has to start by manually finding the minimum and maximum valley widths , the minimum and maximum ridge widths and also the minimum radius of curvature of a ridge . This is not very difficult if the image is zoomed by a very large factor . These parameter are used for finding the size and the coefficients of the enhancement mask . Typical window size for enhancement mask is 13×13 , whereas it is 11×11 for the mask used for finding the prominent direction . This method is very much sophisticated and accurate but its major disadvantage is its complexity and cost of computation .

Mahtre' s method [3]

This method also uses a directional image of the fingerprint for segmentation . Here the directionality calculation is not over the entire window but in specific directions only . More specifically the direction at point (i,j) is defined by the following equation :

$$dir(i,j) = Min \sum_{\ell=1}^{n} (c(i_\ell, j_\ell) - c(i,j))$$

Where $c(i,j)$ and $c(i_\ell, j_\ell)$ indicates the gray scale of the points (i,j) and there extrapolation along a particular direction respectively . Here n is the number of pixels chosen for this computation . Only eight directions are considered here . Now the image is segmented into foreground and background by taking the histogram of the directional Image in a 16×16 window and observing whether the peak exceeds a particular threshold . If the threshold is exceeded then the region is in the foreground otherwise it is in the background .

Another criteria used is the variance of the histogram values . If the variance is higher than another threshold then the pixel belongs to the foreground . On the other hand if the difference between maximum and minimum values in the histogram is less than some another threshold then the pixel belongs to the background .

This method is quite simple and easily implementable but its performance is very much dependent on the quality of the input image . Here a number of thresholds need to be selected and the quality of the output depends on these values . In the case of noisy images some improvement is done by performing an initial averaging with a 3×3 mask over the entire image . The method gives poor result for images with wide and jagged ridges .

4 PROPOSED METHOD

The raw image is a gray tone picture with real and spurious imperfections . Our aim is to retain minutiae and eliminate the spurious imperfections , also the final result is a two tone version in which ridges and valleys are clearly demarcated . Segmentation of the initial gray tone image will produce erroneous results , because binarization of the initial image will result in loss of lot of information features . So here we do some processing on the input gray level image , which pushes up the pixels belonging to background and pulls down others that belongs to the object . In the following step we do ridge crispening . Ridge crispening is an operation in which pixels on the periphery of a ridge are pushed towards the valley by putting local maxima at the point . Effectively in this step ridges are shrinked and a result after few iteration of ridge crispening we are left with the core region of the ridges . Image resulting from this step contains a number of spurious breaks as well as ridge jaggedness . These defects are removed in the subsequent steps . Histogram of the image after above processing becomes bipolar and hence is segmented into object (ridges) and background (valleys) by histogram binarization . This binarized image contains extra wide ridges and hence is thinned .

Proposed method comprises of following steps :

1. Enhancement .
2. Ridge crispening .

3. Filling & Connection of broken ridges .

4. Smoothing away the projection from ridges .

5. Binarization .

6. Thinning .

4.1 Enhancement

The raw inputs are generally gray level image of low contrast and might contain a lot of noise . It is firstly enhanced in the frequency domain by using histogram equalization technique . The resultant output is fed through a directional filter . Where at first we compute the prominent direction at a candidate pixel position . The direction at a point is determined by computation over a 7×7 window centered at that point . Here we consider 12 directions . The prominent direction at point (i, j) is determined by computing the absolute sum of successive difference of pixel gray value in all directions . As for example sum for horizontal direction may be calculated as follows .

$$Sum0 = \sum_{l=j-3}^{j+2} (a_{i,l} - a_{i,l+1})$$

In the Similar way sum for remaining 11 directions are computed . Now prominent direction is one along which this sum is minimum . In case of a tie involving two directions any one is chosen as the prominent direction . If more than two direction has this minimum value then the pixel is said to have random orientation . If a pixel has a definite direction then the average of pixel gray level in the prominent direction is computed . This value is compared with the eight neighborhood average . If eight neighborhood average is less then it is assigned to the candidate pixel otherwise assignment is done with directional average . Image thus filtered is contrast stretched . Several iterations of sharpening and averaging is done to improve the result .

Sharpening is performed by convolving the whole Image with a mask of size 3×3 as shown in fig. 6 .

Averaging is performed by assigning to the candidate point the average gray value computed over a 3×3 window centered at that point . Averaging is thus performed by convolving the whole image with a 3×3 matrix as shown in fig. 7 .

One final iteration of the sharpening is required to complete the enhancement step .

4.2 Ridge crispening

Crispening of ridges is done in the gray level by using an algorithm similar to one proposed by kundu et al . Here at a candidate pixel we consider a window as shown in the figure 8 . The pixels in this window are temporarily binarized . If the candidate point has random orientation then we take eight neighborhood average as the threshold , else we determine threshold in the following way. We consider a 7×7 window at the candidate point and computed average in all the 12 directions and then define threshold as

$$threshold = 0.7\ avg1 + 0.3\ avg2$$

where $avg1$ is average of directional average in directions , which are in the range +45 degree to −45 degree with respect to the prominent direction .$avg2$ is the average of remaining directional averages .

Each iteration of this algorithm has two passes . In the first pass contour point p1 is deleted from the digital pattern if it satisfies following conditions .

(a) $3 \leq B(p_1) \leq 6$.

(b) $A(p_1) = 1$

(c) $p_2 * p_4 * p_6 = 0$

(d) $p_4 * p_6 * p_8 = 0$

Where $p_2, \ldots p_9$ etc are different pixel positions with respect to the candidate pixel p_1 as shown in Fig. 8 .

Here $A(p_1)$ is the no. of 0 1 transition in the ordered sequence $p_2, \ldots, p_9$ and $B(p_1)$ is the no. of non-zero neighbors of p_1 . In the second pass conditions (c) & (d) are changed , and the changes are as follows :
(c') $p_2 * p_4 * p_8 = 0$
(d') $p_2 * p_6 * p_8 = 0$

For avoiding end erosion etc. we introduced another constraint , which is $(p_{10} * p_{11})$ OR $(p_{11} * p_{12})$ OR $(p_{12} * p_{13})$ OR $(p_{13} * p_{10}) = 0$

4.3 Step 3

Step 2 is repeated till desired result is obtained . In each iteration of step 2 we do thresholding of wider margin of gray level values from the brighter gray level range to the background and darker gray level range to the object . The margin grows with each iteration , but is limited to a maximum of 0.25 times the total gray level range.

4.4 Filling

In this step one pixel wide spurious breaks are removed by using eight neighborhood information . A break is removable if any one of the following conditions holds

(a) $A(p_1) = 2$ and
$B(p_1) = 2$ and
isolated zero = 0 and
(zeroes in the 8 th neighborhood
which are sandwiched between two ones)
plus either of these two condition sets

(i) corner black pixels < 2 or
(ii) corner black pixels= 2 and
$(p_{11} * p_3 * p_{12})$ OR $(p_{12} * p_5 * p_{13})$ OR $(p_{13} * p_7 * p_{10})$ OR $(p_{10} * p_9 * p_{11}) = 0$

(b) $A(p_1) = 1$ and
$B(p_1) > 4$

(c) $A(p_1) = 2$ and
$B(p_1) = 4$and
isolated zeroes = 1 and
corner black pixels= 2

(d) $A(p_1) = 2$ $B(p_1) > 4$

4.5 Connection

Connection is done if there are two dangling ridges with there end points forming a line , whose slope is same as the prominent direction in that region .Here break should be atmost 3 pixel wide . A search space in the prominent direction at the candidate point is created by temporarily binarizing the pixels in the direction . Search space for zero degree is shown below :

$$* \ * \ * \ X \ * \ * \ *$$

Here X is the candidate pixel point .

Following condition should be fulfilled for connection to be done .

Candidate pixel is connected to a dark pixel on one side of the search space
&
On the other side of the search space there is atleast one dark (object) pixel .

4.6 Smoothing

The projection from ridges which are not in the prominent direction are removed . Here it is also verified that there are no dangling ridge end in the vicinity of the candidate pixel , in its prominent direction .

Smoothing is done if following conditions are met .

(a) Candidate pixel is dark .

(b) No dark pixels on the both sides of search space considered .

(c) There is a definite prominent direction .

(d) The candidate pixel is not adjacent to a dark pixel in the prominent direction .

Search space is constructed by temporarily binarizing the pixels in the prominent direction . Search space used here is similar to one used in connection .

4.7 Binarization

Histogram of the Image so obtained is bipolar in nature and hence binarization based on histogram of the image suffices the purpose .

4.8 Two tone thinning

This step is included for thinning extra wide ridges . The thinning algorithm due to Kundu et al , which was used in ridge crispening , is used here .

5 Algorithm & Implementation

The overall Software chart is shown in Fig. 5 . Output of each of the intermediate step can be taken . From the flow chart it is evident some blocks are executed in loop . The number of iteration for these loops are decided subjectively by judging the quality of the image produced . Generally two iteration of the bigger loop is sufficient to produce good result . In each such iteration smaller loop is executed atmost three to four times . The complexity of each step is O(n) . Hence it is executable in linear time . The program has been written in C and is developed on VAX . The is portable to pc environment . The memory limitation of pc restricts the whole package to run at a time . By proper segmentation of code it can however be executed there .

6 Result & Discussion :

Some of the results with different input images are attached . Each separate input has eight images associated with it .

1. Raw image .
2. Output of directional filter .
3. Final enhanced image .
4. Image after ridge crispening .
5. Output of Filling & connection .
6. After smoothing .
7. Binarized image .
8. Image after final thinning .

The raw images are low contrast gray level images containing lot of protruding object pixels . These are mostly removed in the direction filter scheme . The ridge boundaries in these images are not very sharp . Hence we required some sharpening. Few iterations of sharpening interleaved with averaging produced well sharpened ridge boundaries . For images like set 1 this step produce some false connection as side effect . The reason for this is that ridges here are quite close to each other and noise pixel concentration in the intermediate valley is very high and as a result averaging results in these spurious connection . The possible modification sought at the enhancement step is to consider the general trend of ridge flow in the vicinity of the candidate point .

These enhanced images contain thick and extra wide ridges . After ridge crispening only the core region of the ridges are left . As a side effect this step introduces some spurious breaks which are removed by our filling and connection step . Image so obtained has some jaggedness which is removed in the following step . The Image after smoothing almost becomes similar to a bilevel image . This is the reason for adopting histogram segmentation to get two tone image . The binary image has some extra wide ridges . This suggested to use few iterations of thinning algorithm .

7 REFERENCES

1. Gonzalez and Wintz , " Digital Image Processing 2nd ed. ".

2. C. V. Kameswara Rao ,"Pattern Recognition Techniques Applied to to Fingerprints", Ph.D thesis , Linkoping University , Sweden .

3. Mehtre , Murty , Kapoor & Chatterjee ,"Segmentation of Fingerprint Images using directional Image ",Pattern Recognition letters , 1987 .

4. L. O'Gorman & J. V. Nickerson ," An Approach to Fingerprint Filter Design",Pattern Recognition Letters , Vol. 22 , No. 1,1989.

5. M. K. Kundu , B. B. Choudhury, and D. Dutta. Majumdar,"A Parallel Gray tone thinning Algorithm (PGTA) ", Pattern Recognition Letters , Vol. 12,PP. 491-495 , 1991 .

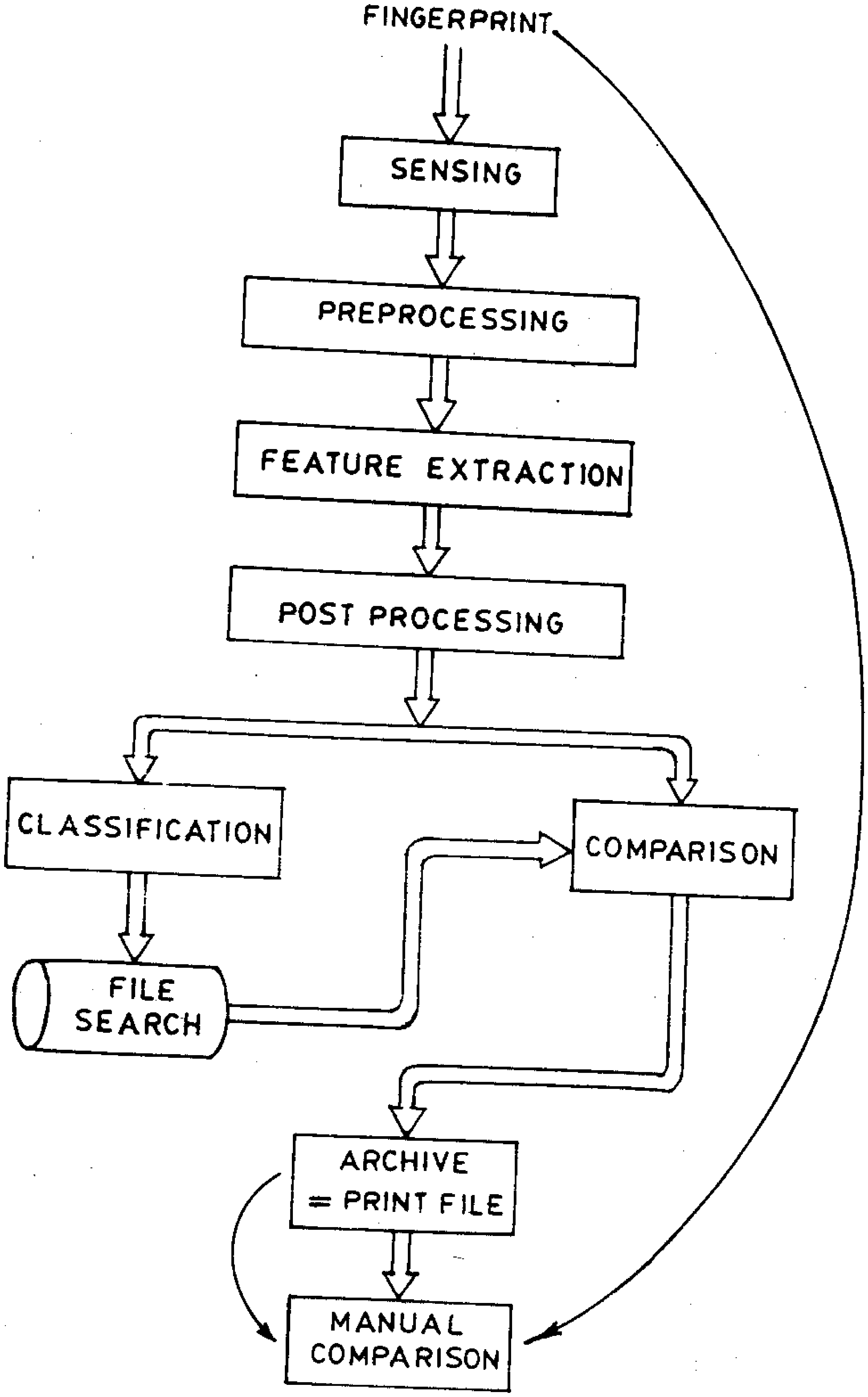
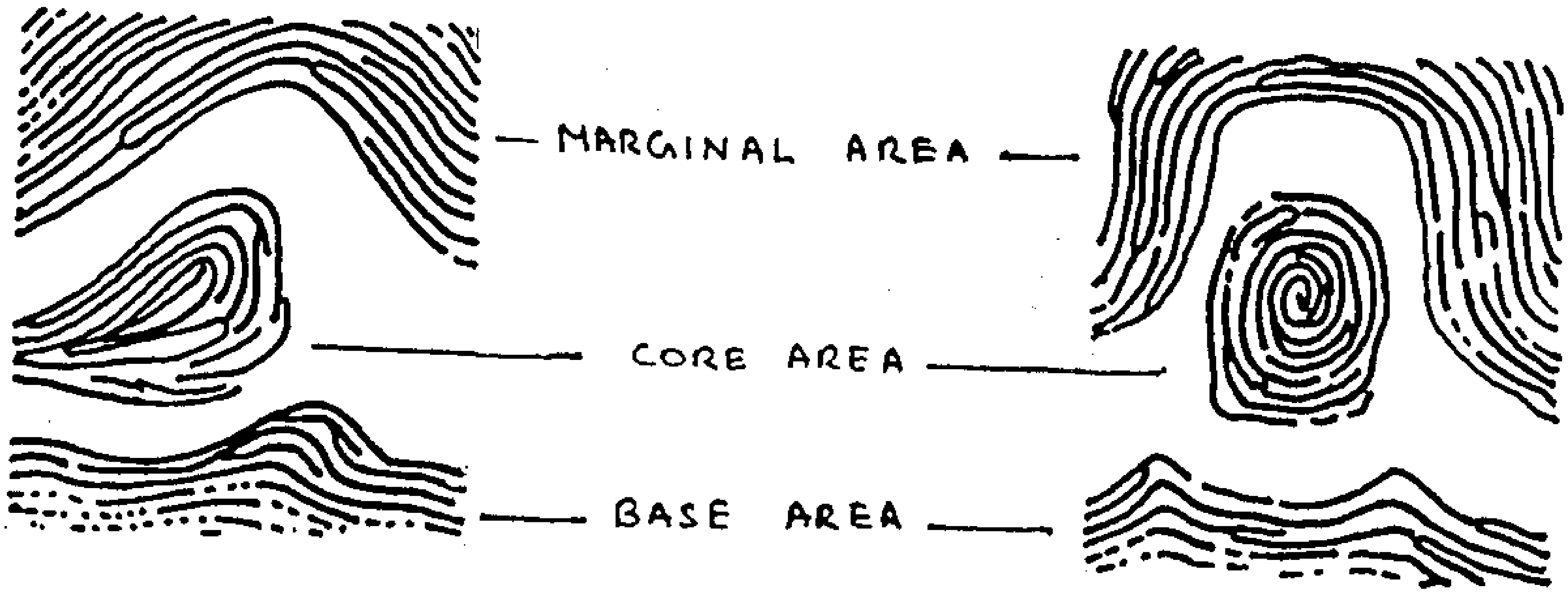
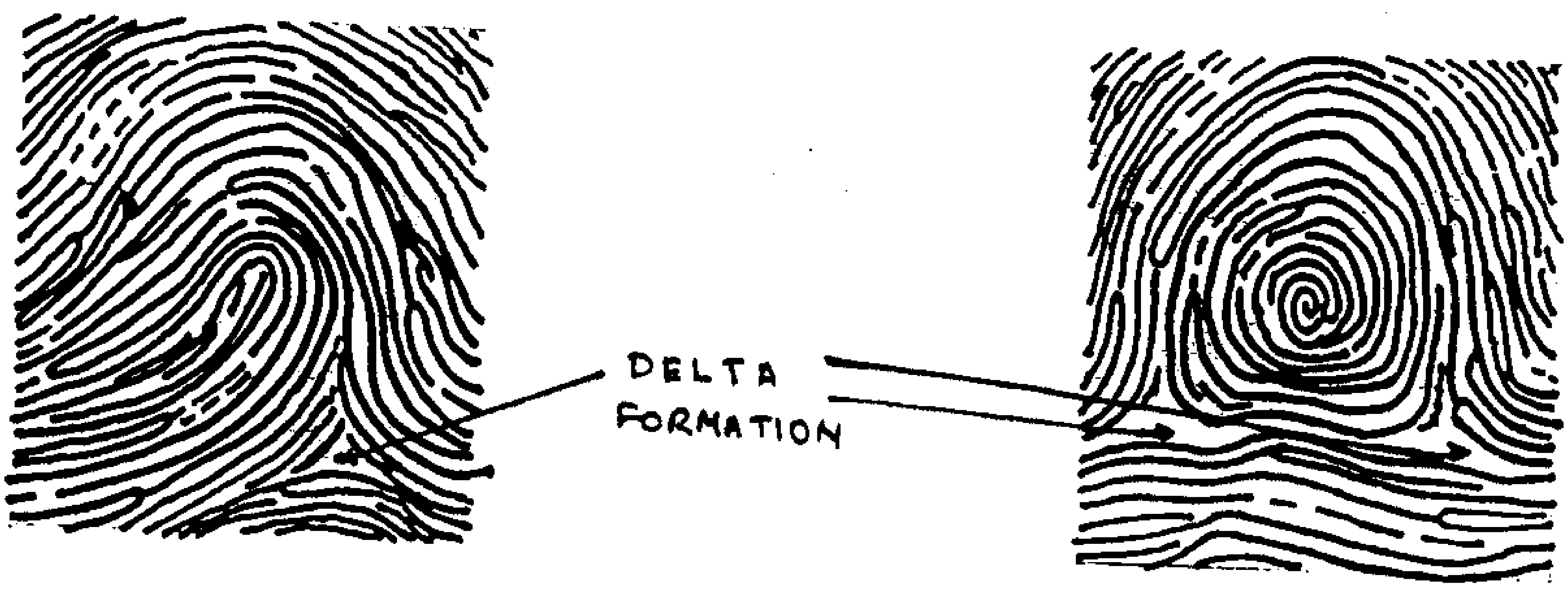


Fig. 1: FLOWCHART DESCRIBING THE GENERAL STEPS IN THE PROCES OF FINGERPRINT IDENTIFICATION.



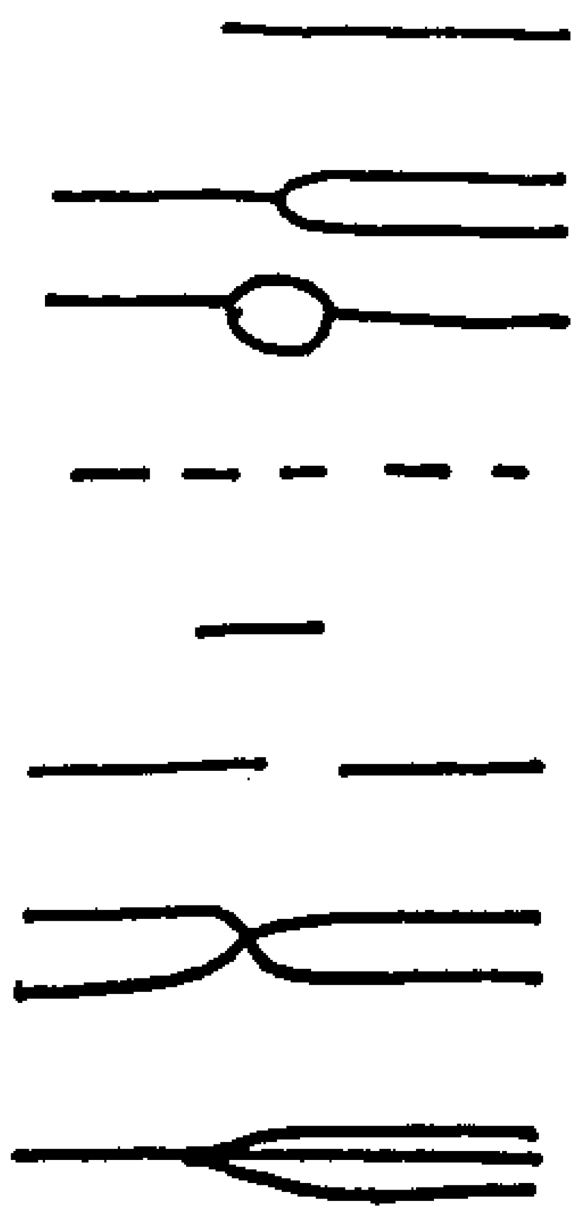
FINGERPRINT DIVIDED INTO VARIOUS AREAS



DELTA FORMATION

Fig -2

DIFFERENT TYPES OF FINGERPRINTS
FIG 3



1. RIDGE ENDING
2. BIFURCATION OR FORK
3. ENCLOSURE
4. ISLANDS OR RIDGE DOTS.
5. SHORT RIDGE
6. RIDGE BREAK
7. RIDGE CROSSING
8. TRIFURCATION.

MINUTIAE.

EXAMPLE . . .

FIG-4

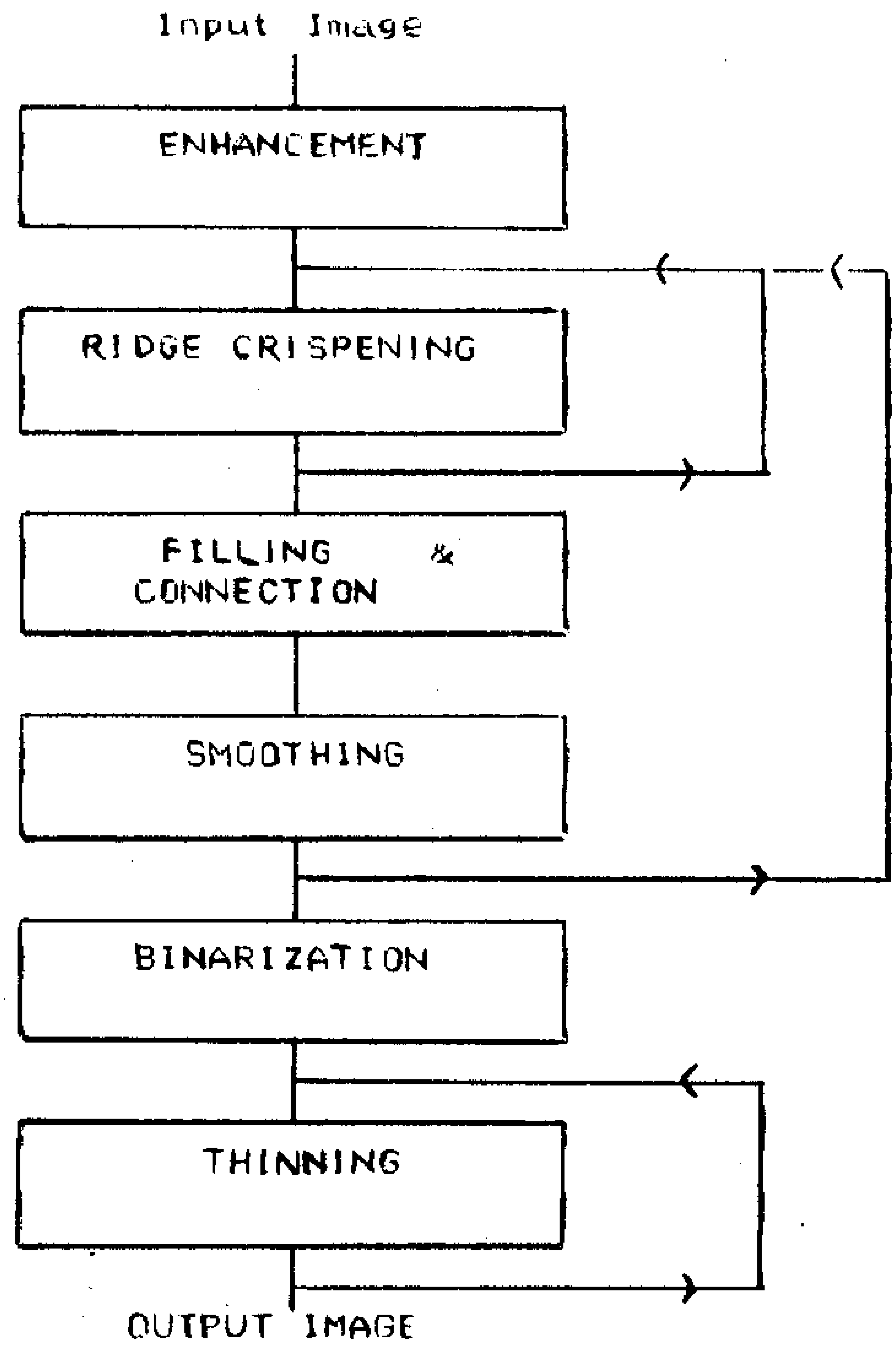

Input Image
ENHANCEMENT
RIDGE CRISPENING
FILLING &
CONNECTION
SMOOTHING
BINARIZATION
THINNING
OUTPUT IMAGE

Fig. 5

	- 1	
- 1	- 5	- 1
	- 1	

Fig. 6

1	1	1
1	1	1
1	1	1

Fig. 7

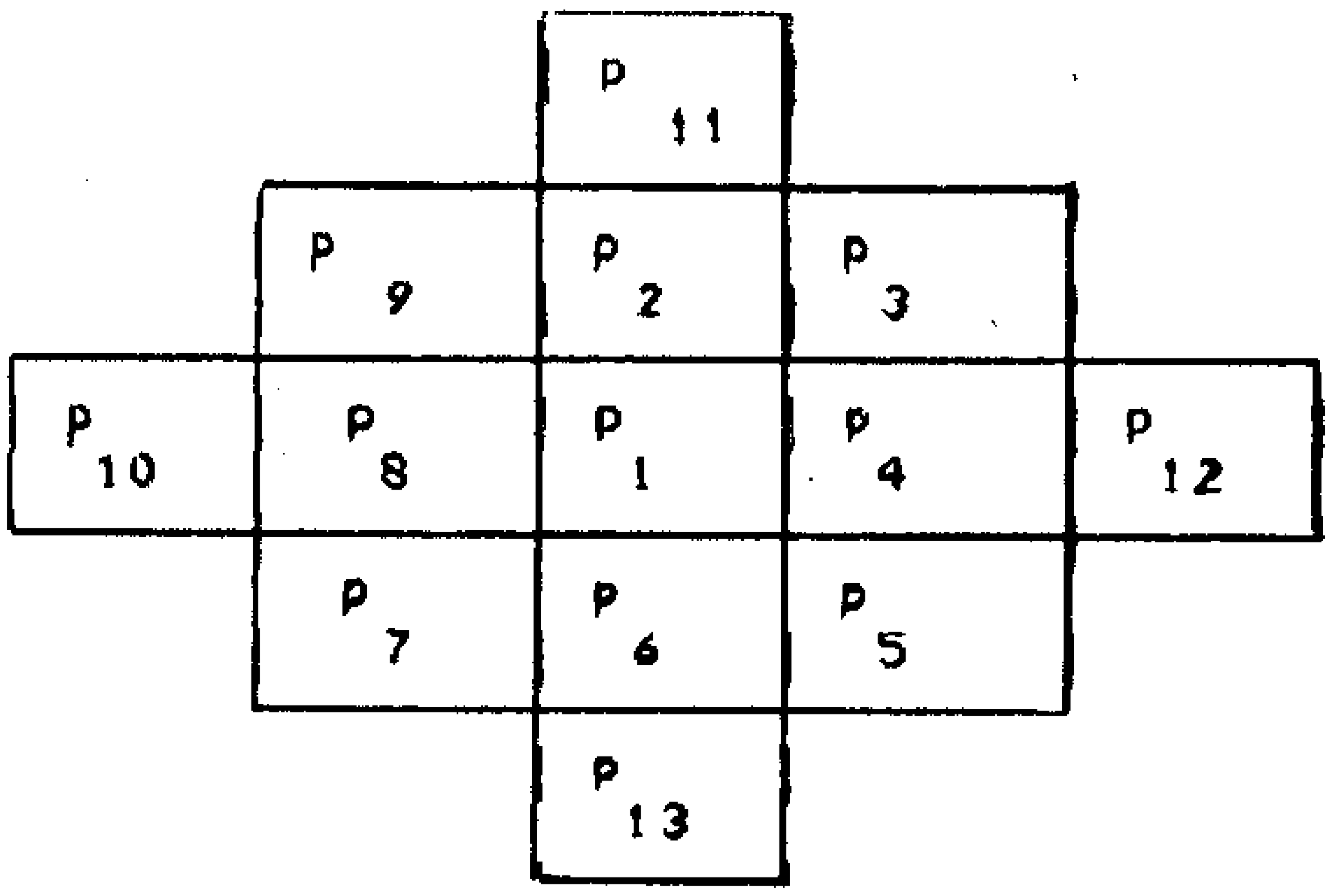
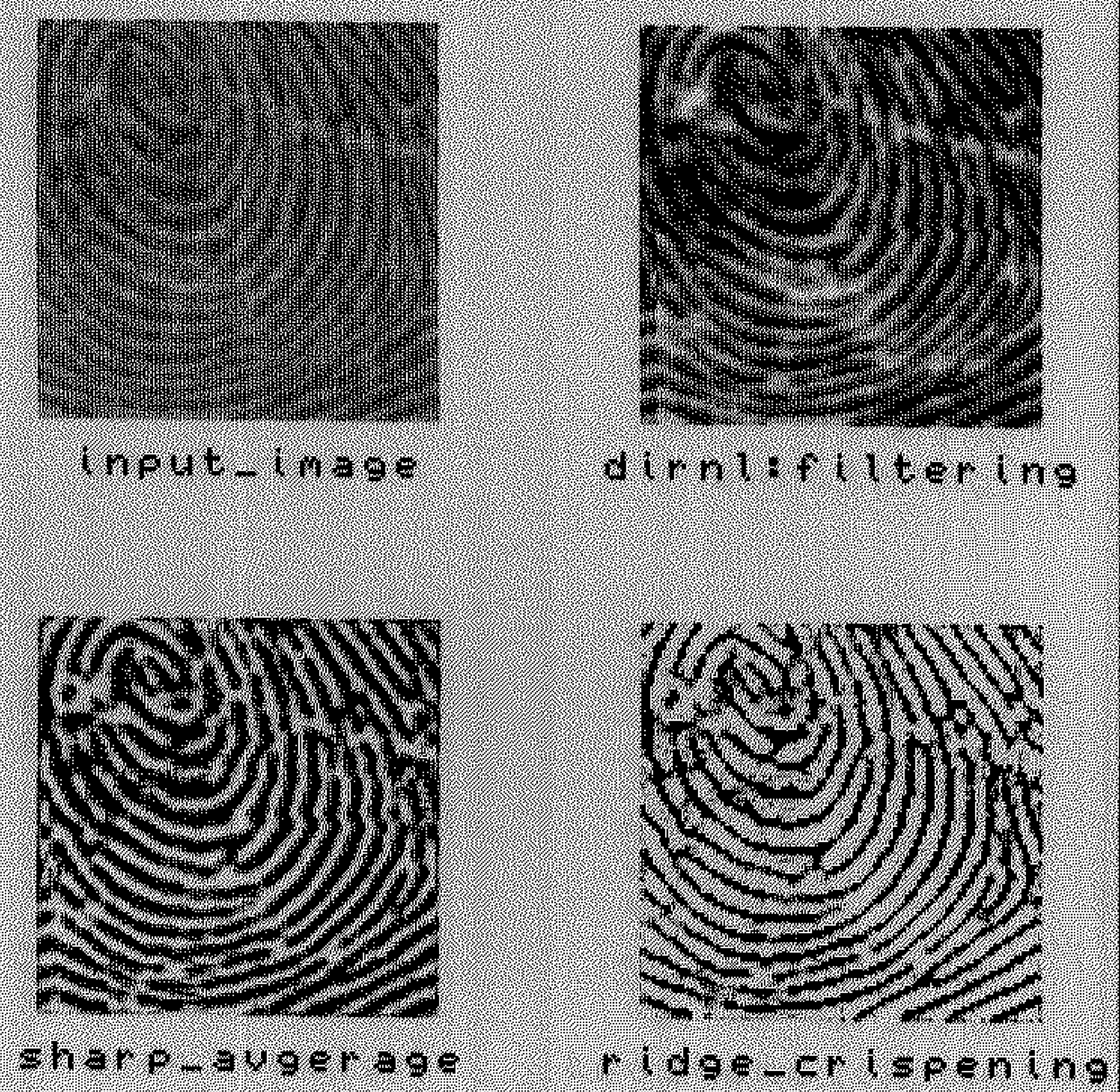
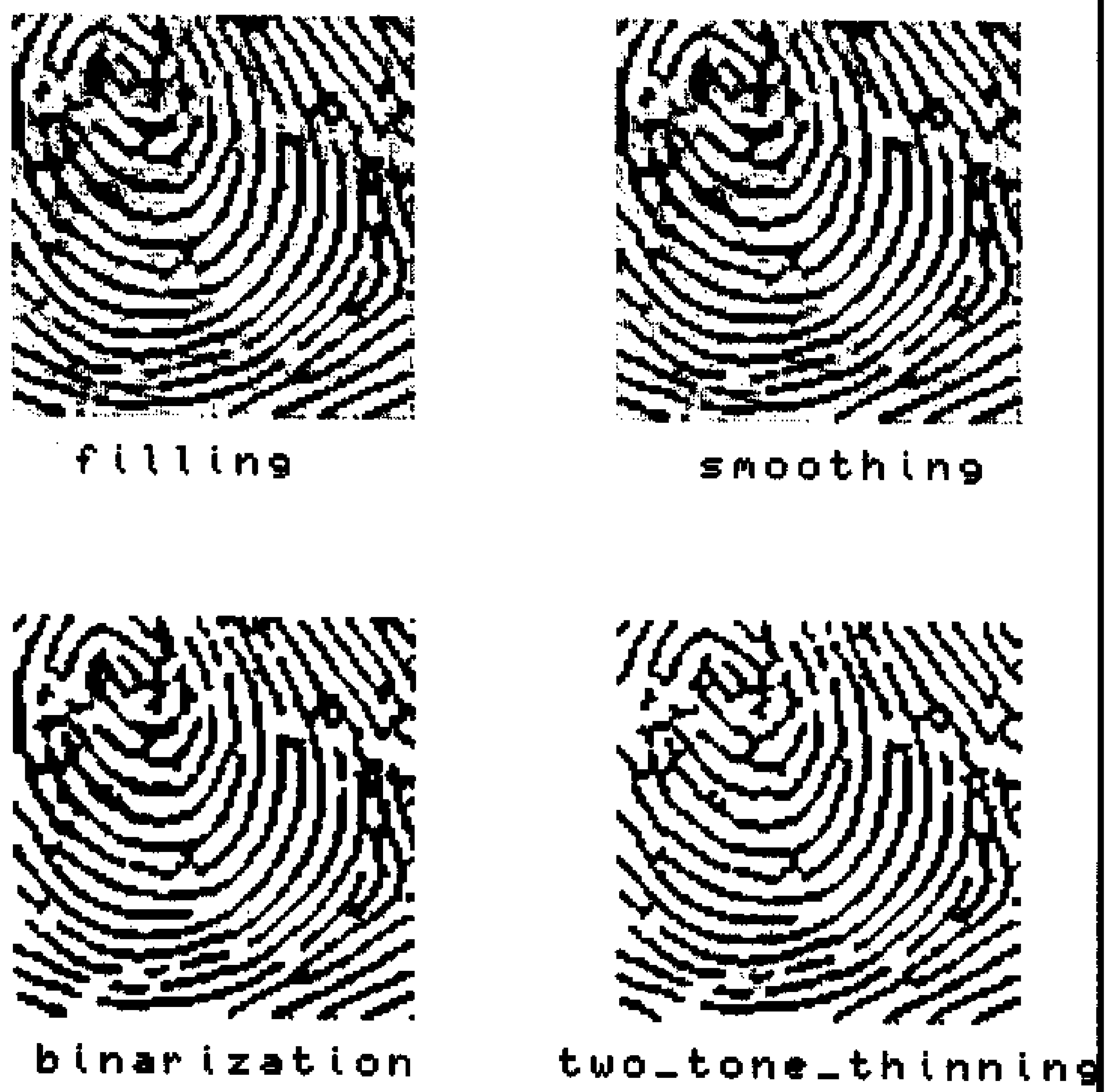


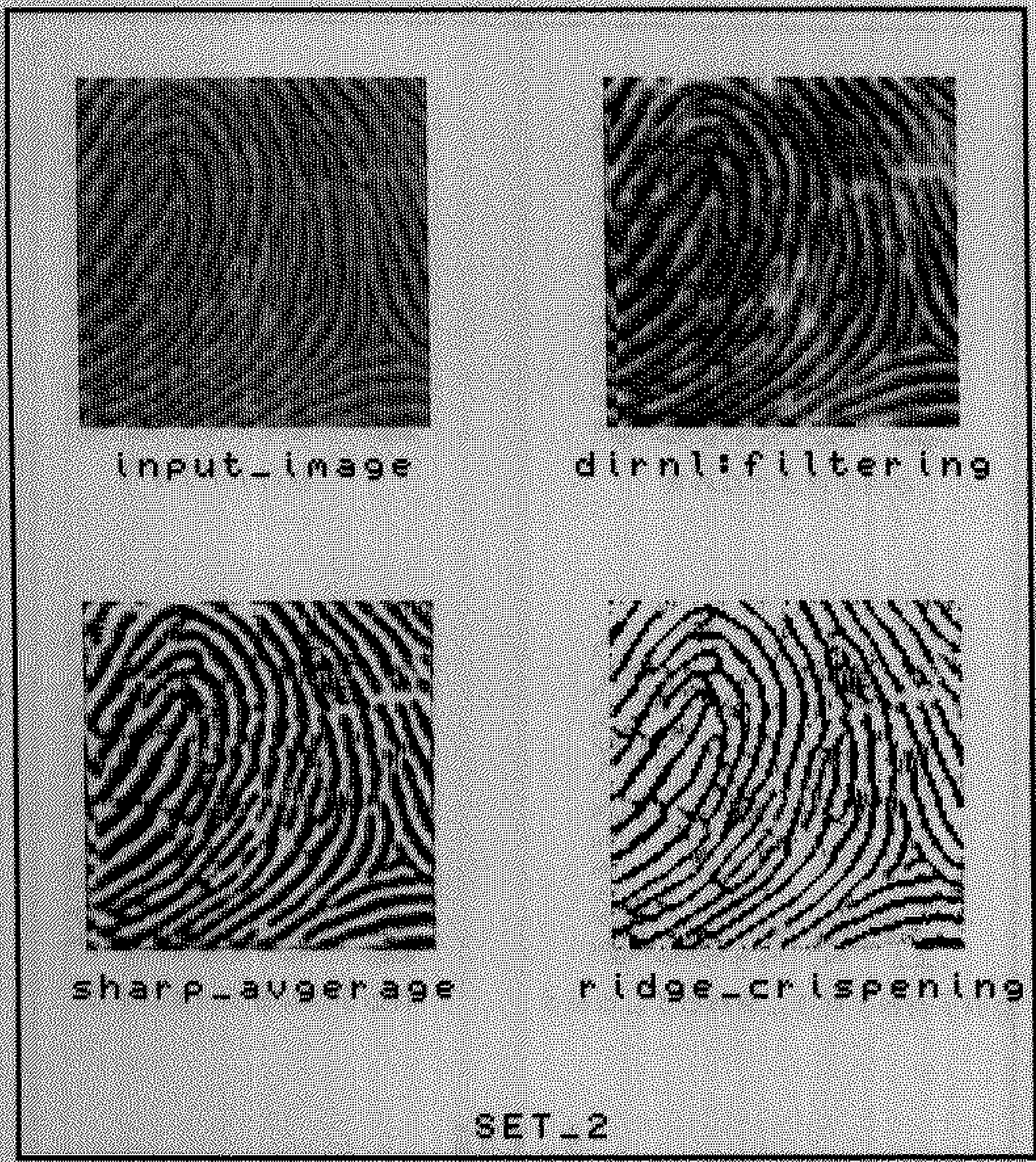
Fig. 8

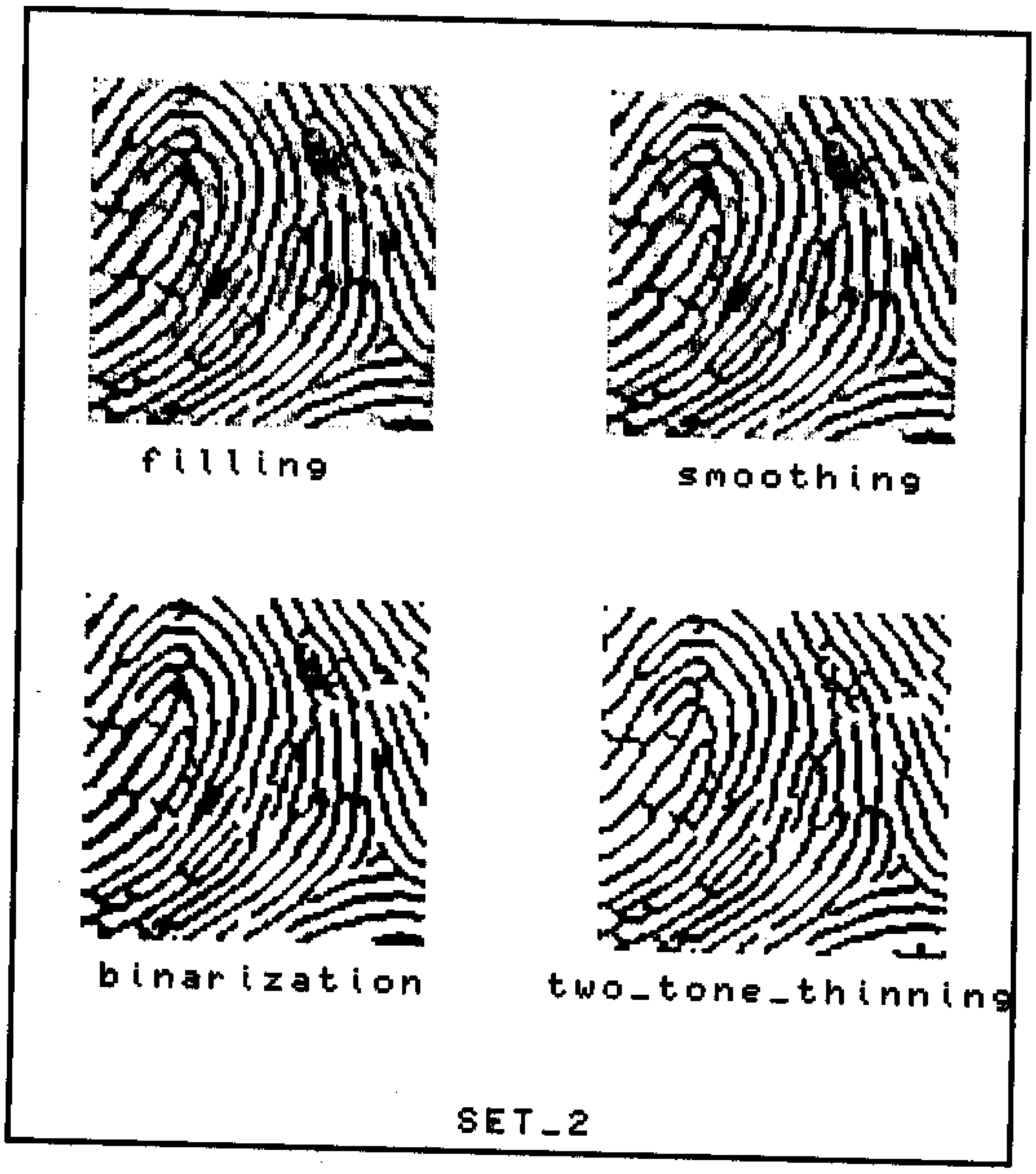

input_image
dirnl:filtering
sharp_avgerage
ridge_crispening

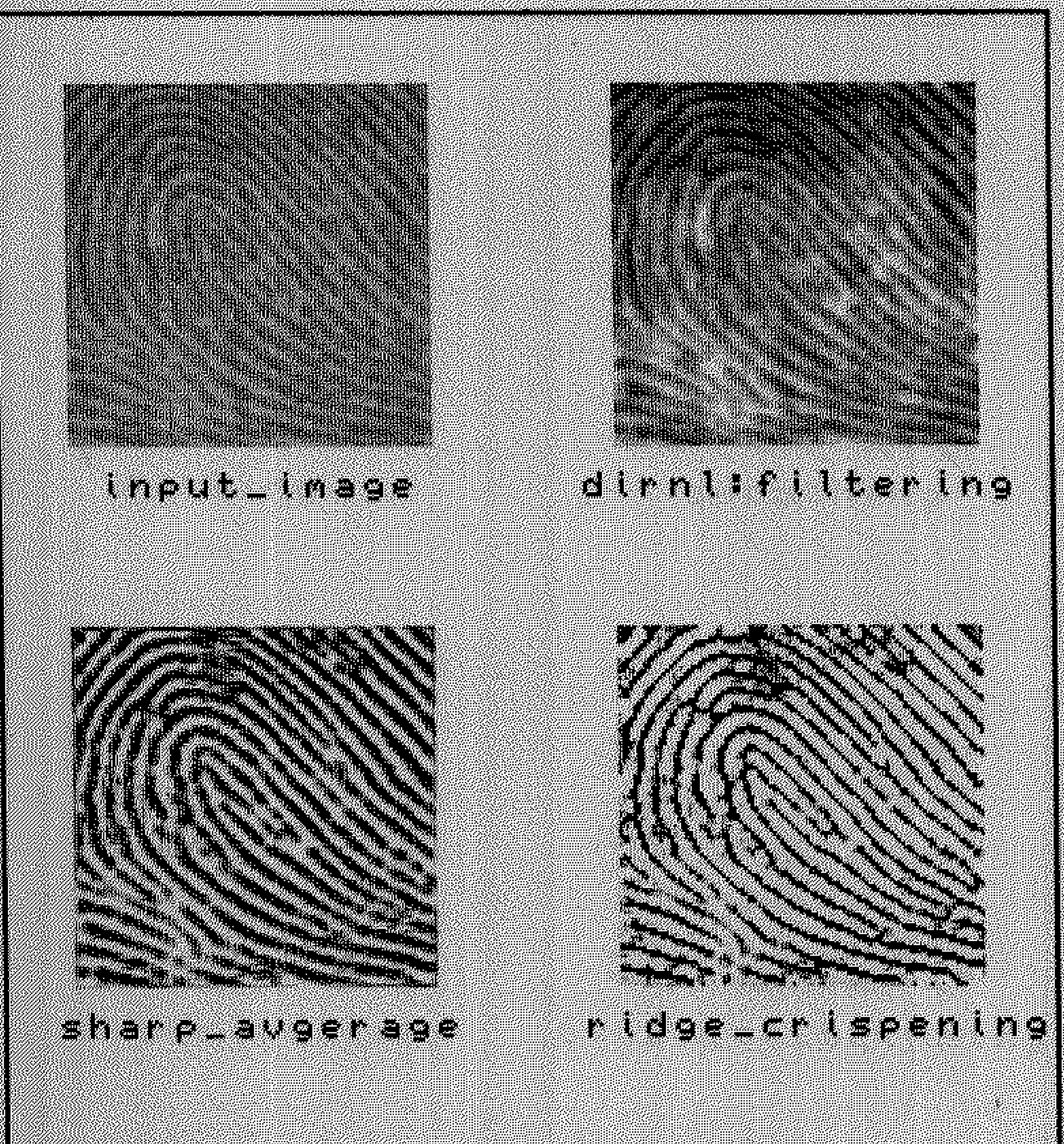
SET_1



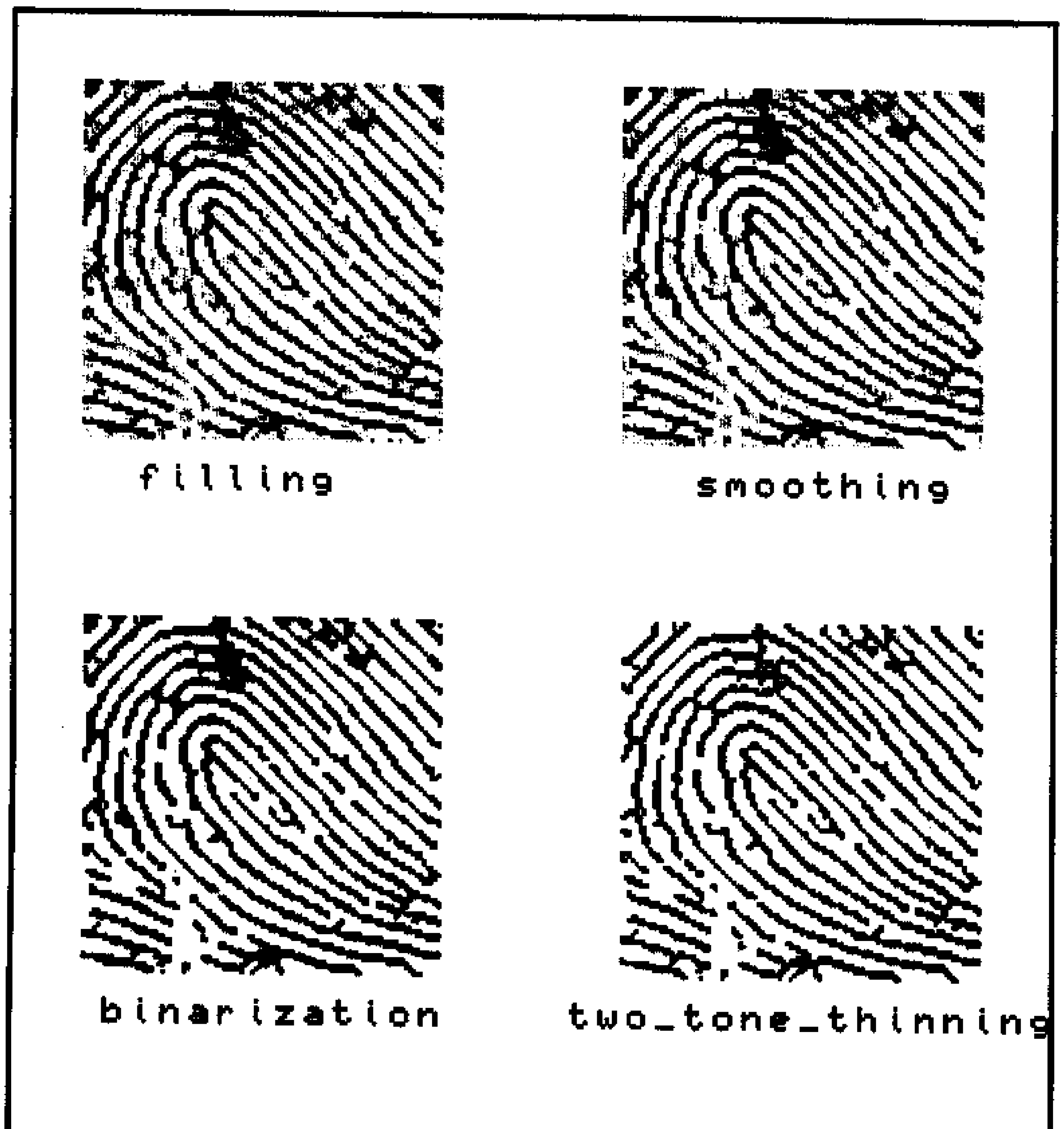
SET_1


input_image
dirnl:filtering
sharp_avgerage
ridge_crispening
SET_2

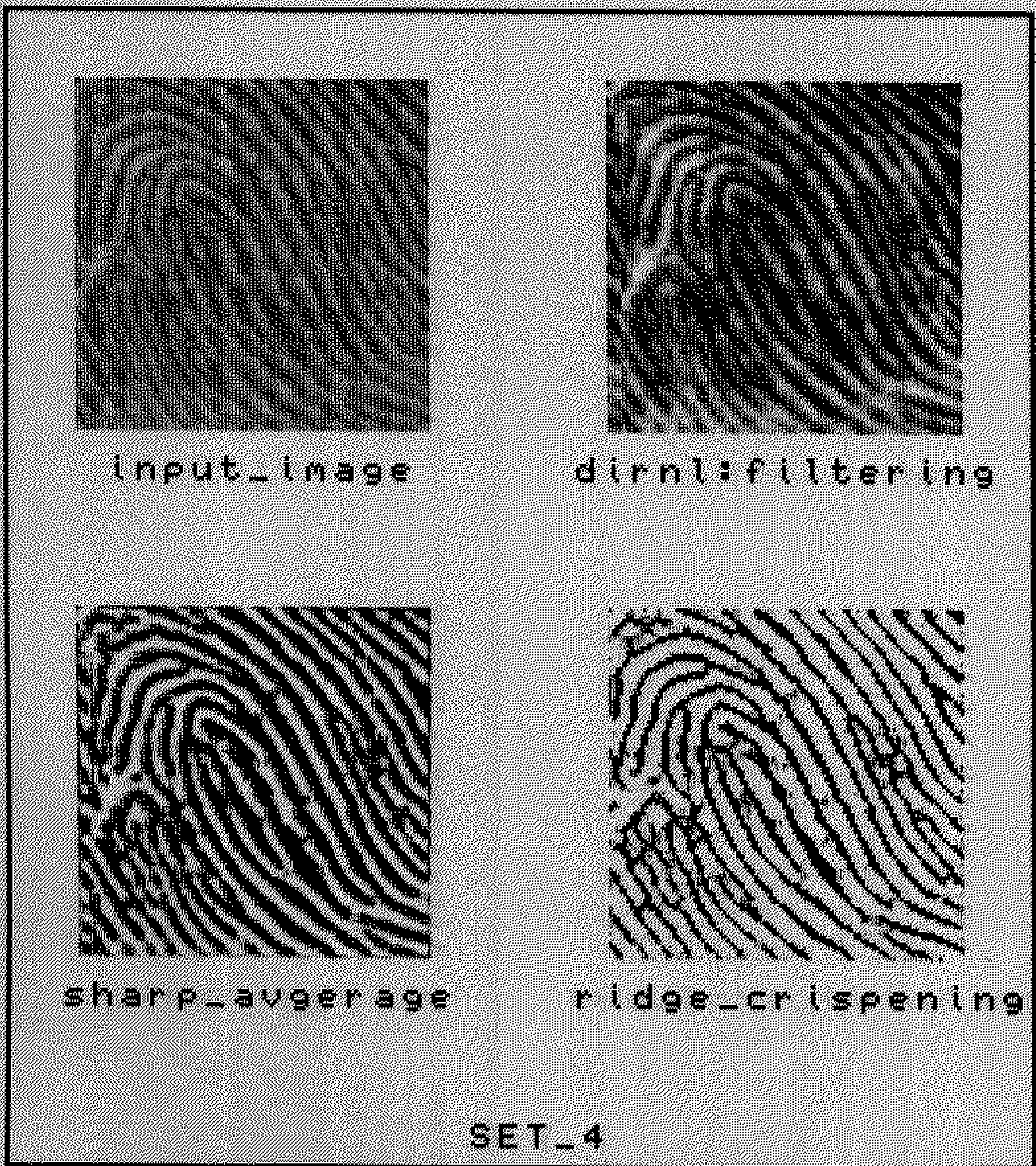

filling
smoothing
binarization
two_tone_thinning
SET_2


input_image
dirnl:filtering
sharp_avgerage
ridge_crispening

SET_3



SET_3


input_image
dirnl:filtering
sharp_avgerage
ridge_crispening
SET_4

filling

smoothing

binarization

two_tone_thinning

SET_4