INTERNATIONAL STANDARD

ISO/IEC 19794-8

First edition
2006-10-01

Information technology — Biometric data interchange formats —

Part 8: **Finger pattern skeletal data**

Technologies de l'information — Formats d'échange de données biométriques —

Partie 8: Données des structures du squelette de l'empreinte

Reference number
ISO/IEC 19794-8:2006(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 19794-8 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 37, *Biometrics*.

ISO/IEC 19794 consists of the following parts, under the general title *Information technology — Biometric data interchange formats*:

— *Part 1: Framework*

— *Part 2: Finger minutiae data*

— *Part 3: Finger pattern spectral data*

— *Part 4: Finger image data*

— *Part 5: Face image data*

— *Part 6: Iris image data*

— *Part 7: Signature/sign time series data*

— *Part 8: Finger pattern skeletal data*

— *Part 9: Vascular image data*

— *Part 10: Hand geometry silhouette data*

— *Part 11: Signature/sign processed dynamic data*

Introduction

With the interest of implementing interoperable personal biometric recognition systems, this part of ISO/IEC 19794 establishes a data interchange format for pattern-based skeletal fingerprint recognition algorithms. Pattern-based algorithms process sections of biometric images. Pattern-based algorithms have been shown to work well with the demanding, but commercially driven, fingerprint sensor formats such as small-area and swipe sensors.

The exchange format defined in this part of ISO/IEC 19794 describes all characteristics of a fingerprint in a small data record. Thus it allows for the extraction of both spectral information (orientation, frequency, phase, etc.) and features (minutiae, core, ridge count, etc.). Transformations like translation and rotation can also be accommodated by the format defined herein.

With this part of ISO/IEC 19794 for pattern-based skeletal representation of fingerprints

— interoperability among fingerprint recognition vendors based on a small data record is allowed;

— proliferation of low-cost commercial fingerprint sensors with limited coverage, dynamic range, or resolution is supported;

— a data record that can be used to store biometric information on a variety a storage media (including but not limited to, portable devices and smart cards) is defined;

— adoption of biometrics in applications requiring interoperability is encouraged.

It is recommended that biometric data protection techniques in ANSI/X9 X9.84 or ISO/IEC 15408 are used to safeguard the biometric data defined herein for confidentiality, integrity and availability.

Information technology — Biometric data interchange formats —

Part 8:
Finger pattern skeletal data

1 Scope

This part of ISO/IEC 19794 specifies the interchange format for the exchange of pattern-based skeletal fingerprint recognition data. The data format is generic, in that it may be applied and used in a wide range of application areas where automated fingerprint recognition is involved.

2 Conformance

A system conforms to this part of ISO/IEC 19794 if it satisfies the mandatory requirements herein for extraction and description of the skeleton described in Clause 6 and the generation of the data record as described in Clause 7.

Since any finger skeletal data extraction and comparison algorithm supporting the described finger skeletal data interchange formats may be used, interoperability testing is of extreme importance, especially for environments in which components of different manufacturers interact.

3 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 7816-6:2004, *Identification cards — Integrated circuit cards — Part 6: Interindustry data elements for interchange*

ISO/IEC 7816-11:2004, *Identification cards — Integrated circuits cards — Part 11: Personal verification through biometric methods*

ISO/IEC 19784-1:2006, *Information technology — Biometric application programming interface — Part 1: BioAPI specification*

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

4.1
biometrics
automated recognition of individuals based on their behavioural and biological characteristics

4.2
biometric algorithm
sequence of instructions that tell a biometric system how to solve a particular problem

NOTE An algorithm will have a finite number of steps and is typically used by the biometric engine (i.e. the biometric system software) to compute whether a biometric sample and template are a match.

4.3
biometric data
biometric sample at any stage of processing, biometric reference, biometric feature or biometric property

4.4
biometric information template
constructed data object in a card containing information needed by the outside world for a verification process

NOTE See ISO/IEC 7816-11.

4.5
biometric reference
one or more stored biometric samples, biometric templates or biometric models attributed to a subject and used for comparison

EXAMPLES Face image on a passport; fingerprint minutiae template on a national ID card; Gaussian mixture model, for speaker recognition, in a database.

4.6
biometric sample
analog or digital representation of biometric characteristics prior to feature extraction process and obtained from a biometric device

4.7
biometric system
automated system capable of
1. capturing a biometric sample from a subject;
2. extracting a biometric feature from that sample;
3. comparing the biometric feature with that contained in the biometric reference;
4. deciding how well they match; and
5. indicating whether or not an identification or verification of identity has been achieved

4.8
biometric template
set of stored biometric features comparable directly to biometric features of a presented biometric sample

NOTE 1 A biometric reference consisting of an image, or other captured biometric sample in its original, enhanced or compressed form, is not a biometric template.

NOTE 2 The biometric features are not considered to be a biometric template unless they are stored for reference.

4.9
bit-depth
number of bits used to represent a data element

4.10
capture
method of taking a biometric sample from the subject

4.11
cell
rectangular region defined by a uniform and non-overlapping division of the image

4.12
closed-set identification
biometric application that ranks the biometric references in the enrolment database in order of decreasing similarity against a presented biometric sample

4.13
comparison
estimation, calculation or measurement of similarity or dissimilarity between biometric sample(s) and biometric reference(s)

4.14
core
singular point in the fingerprint, where the curvature of the ridges reaches a maximum[1)]

NOTE For simplicity, the core can be considered as a U-turn, sometimes enclosing a few ridge endings. It serves as an approximation of the centre of the fingerprint image.

4.15
delta
structure where three fields of parallel ridge lines meet[1)]

NOTE From Danuta Z. Loesch, “Quantative dermatoglyphics – classification, genetics, and pathology”, Oxford Monographs on Medical Genetics No. 10, Oxford University Press 1983, ISBN 0-19-261305-7, page 7.

4.16
dimension
number of pixels in an acquired biometric sample in either the x- or y- direction

4.17
enrolment
process of creating and storing, for an individual, a data record associated with an individual and including biometric reference(s) and, typically, non-biometric data

4.18
friction ridge
structure on the skin of the fingers and toes, the palms and soles of the feet, which makes contact with an incident surface under normal touch

NOTE On the fingers, the unique patterns formed by the friction ridges make up fingerprints.

4.19
identification
biometric system function that performs a one-to-many search

NOTE An identification function may be used to verify a claim of enrolment in an enrolment database without a specified biometric reference identifier.

4.20
latent
fingerprint collected from an intermediate surface, rather than directly via a live capture from the finger itself

4.21
live capture
process of capturing a biometric sample by an interaction between a subject and a biometric system

[1)] The definitions of core and delta in ISO/IEC 19794-3 and this part of ISO/IEC 19794 are identical. However there is a different definition in ISO/IEC 19794-2. Although both definitions try to define the same thing, this difference has occurred for historical reasons.

4.22
minutia
friction ridge characteristic, occurring at a point where a single friction ridge deviates from an uninterrupted flow, that is used to individualize a fingerprint

NOTE 1 Deviation may take the form of ending, division, or a more complicated "composite" type.

NOTE 2 The plural of minutia is minutiae.

4.23
one-to-many search
comparison process in which a biometric sample set of one individual is compared against the biometric references of more than one individual to return a set of comparison scores

NOTE 1 A biometric identification function performs a one-to-many search.

NOTE 2 In the case of a multimodal biometric system, biometric sample and biometric reference in the above definition comprise individual biometric samples/references of the component modalities.

NOTE 3 The degree of similarity may be specified on the basis of comparison score and/or rank.

4.24
open-set identification
biometric application that determines a possibly empty candidate list by collecting one or more biometric samples from an individual and searching the enrolment database for similar biometric references

4.25
record
reference and other information about the subject

NOTE E.g. to access permissions.

4.26
resolution
number of pixels (picture elements) per unit distance in the image of the fingerprint

4.27
ridge bifurcation
minutia assigned to the location at which a friction ridge splits into two ridges or, alternatively, where two separate friction ridges combine into one

4.28
ridge ending
minutia assigned to the location at which a friction ridge terminates or begins

4.29
skeleton
line representation of an object that is one pixel thick through the "middle" of the object and preserves the topology of the object

4.30
swipe
method of fingerprint collection where the finger is manually moved across a one-dimensional sensor to produce the two-dimensional image

4.31
sweat pore
minute opening in the dermis, allowing loss of fluid as a part of the temperature control of the body

4.32
user
client to any biometric vendor

NOTE The user must be differentiated from the end-user (subject) and is responsible for managing and implementing the biometric application rather than actually interacting with the biometric system.

4.33
valley
area surrounding a friction ridge, which does not make contact with an incident surface under normal touch

4.34
verification
verify
process of comparing a submitted biometric sample against the biometric reference template of a single enrolee whose identity is being claimed, to determine whether it matches the enrolee's template

cf. **identification**

5 Abbreviated terms

For the purposes of this document, the following abbreviated terms apply.

BER	Basic Encoding Rules
BIT	Biometric Information Template
CBEFF	Common Biometric Exchange Formats Framework
DO	Data Object
ppcm	pixels per centimetre

6 Determination of finger pattern skeletal data

This ISO/IEC standard for finger pattern interchange data is based on the skeleton representation of friction ridges. Since the result of different skeleton generation algorithms will differ at a maximum of about a quarter of the ridge width this will have no impact on interoperability. In order to get a robust skeleton of the ridges a noise reduction and regularization may take place on the raw image. The direction encoding of the skeleton line elements is included in the interchange data record. The start and endpoints of the skeleton ridgelines are included as real or virtual minutiae, and the line from start to endpoint is encoded by successive direction changes. In the following first the minutiae characteristics and then the encoding definition for one skeleton line is described.

6.1 Minutia

Minutiae are points located at the places in the fingerprint image where friction ridges end or split into two ridges.

6.1.1 Minutia type

Each minutia point has a “type” associated with it. There are two major types of minutia: a “ridge ending” represented by the 2-bit value 01 and a “ridge bifurcation” or split point represented by 2-bit value 10. Points with three or more intersecting ridges (trifurcations, etc.) will be treated as a “ridge bifurcation” type.

Ridge skeletons require the use of both real and "virtual" minutiae. Virtual minutiae are points on the fingerprint image where a real ridge ending or a bifurcation does not exist, but a point is required to finish, or continue, a skeleton ridgeline. Virtual minutiae have thus two types: virtual endings and virtual continuations.

— Virtual endings are necessary to describe skeleton lines ending at the image boundary or at border lines to those areas where there is insufficient image quality to determine ridges and real minutiae points (see Figure A.3). They are also needed to finish the encoding of a closed loop (Table A.1). Virtual endings have been assigned the 2-bit value 00.

— In rare cases a skeleton line description will require the insertion of a virtual minutia point on a ridgeline. For example, such points will be required to begin an encoding of a closed loop for which no real minutiae exist, as well as to describe ridges with high curvature at a sufficient accuracy (see note about maximal curvature in 6.2.4). These are called "virtual continuation" and have been assigned the 2-bitvalue 11 (Table A1).

6.1.2 Minutia location and coordinate system

The coordinate system used to express the position of the minutiae points of a fingerprint shall be a Cartesian coordinate system. Points shall be represented by their x and y coordinates, where x increases to the right and x increases downward (opposite of the pointing direction of the finger), when viewing on a latent print of the finger (see Figure 1). Note that this is in agreement with most imaging and image processing use. When viewing on the finger, x increases from right to left as shown in Figure 1. All x and y values are non-negative. For the skeletal pattern record format, the resolution is specified in the record header, see 7.3.7. For the skeletal pattern card format, the resolution of the x and y coordinates of the minutia shall be in metric units. The granularity is one bit per five hundredth of a millimetre in the normal format and one tenth of a millimetre in the compact format:

1 unit = 0,05 mm (normal format) or 0,1 mm (compact format).

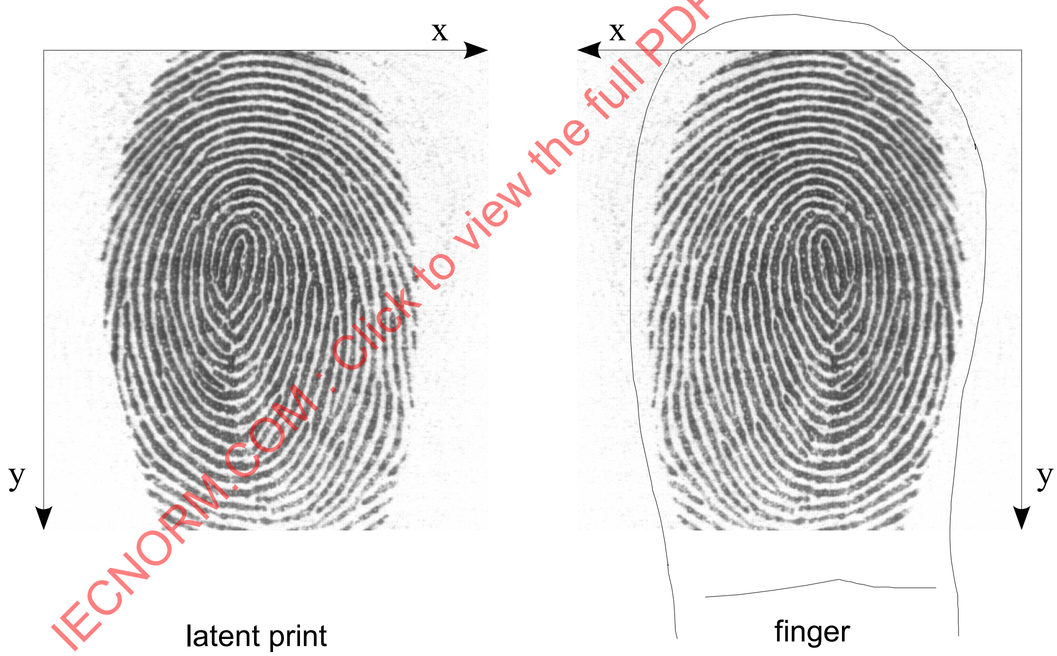


Figure 1 — Coordinate system

The position of the minutia for a ridge ending shall be defined as the coordinates of the skeleton point with only one neighbour pixel belonging to the skeleton.

NOTE In some format types of ISO/IEC 19794-2 a ridge ending refers to the point of bifurcation of the valley in front of the ridge.

The position of the minutia for a ridge bifurcation shall be defined as the point of forking of the skeleton of the ridge. In other words, the point where three or more ridges intersect is the location of the minutia.

The position of a virtual ending shall be defined like the position of a real ridge ending.

The position for the minutiae type “virtual continuation” is not evaluated by comparison algorithms, that analyse minutiae points and angles only. Minutiae of this type are only used for reconstructing the skeleton but may support subsequent classifications of the reconstructed pattern. One may assign any point on the skeleton necessary to increase the accuracy of the ridge line description (Table A.1).

6.1.3 Angle conventions

The minutiae angle is measured increasing counter clockwise starting from the horizontal axis to the right. The angle of a minutia is scaled to fit the bit width of the data field defined in the record header.

The direction of a ridge skeleton endpoint is defined as the angle between the tangent to the ending ridge and the horizontal axis extending to the right right of the ridge ending point.

A ridge skeleton bifurcation point has three intersection ridges. The two ridges enclosing the ending valley encompass an acute angle. The direction of a ridge bifurcation is defined as the mean direction of their tangents. Where each direction is measured as the angle the tangent forms with the horizontal axis to the right.

The direction of the lines starting or ending at a point with more than three arms (trifurcation, etc.) shall be defined like the direction of a real ridge ending.

The direction of a virtual ending shall be defined like the direction of a real ridge ending.

The direction for the minutia type “virtual continuation” is not evaluated by comparison algorithms, that analyse minutiae points and angles only. Minutiae of this type are only used for reconstructing the skeleton but may support subsequent classifications of the reconstructed pattern. One may assign the mean of the incoming and outgoing direction or the outgoing direction (Table A.1).

6.1.4 Differences to minutia data in ISO/IEC 19794-2 – finger minutia data

The definition of the minutia position and direction is identical with ISO/IEC 19794-2 card format (Format type ‘0004’ or ‘0006’) with

— minutia placement on a ridge bifurcation encoded as a ridge skeleton bifurcation point, and

— minutia placement on a ridge skeleton endpoint.

To compare minutiae with any other definition, a position and direction correction may be necessary. There may be performance interoperability differences with the other format types of ISO/IEC 19794-2.

The angular resolution of minutiae in the finger pattern skeletal data record is defined in the header. The minimal resolution allowed is 16 directions, that is 22,5° per least significant bit. A resolution below the recommended 64 directions (5,625°)(Table 5: Bit-depth of direction code start and stop direction) may cause a decrease in match quality for purely minutiae based comparison algorithms. This recommendation corresponds to the angular resolution of the compact card format in finger minutiae data.

There are no virtual minutiae (type ID 00 and 11) in the finger minutiae data format.

There is no minutia type “other” (type ID 00) in the skeletal pattern data format.

Point with more than three arms (trifurcation, etc.) are not mentioned in the finger minutiae data, so the may be omitted or encoded as “other”. In the finger pattern skeletal data these structures get the type “bifurcation”.

6.2 Encoding the skeleton ridge line by a direction code

6.2.1 Direction code

Each line in the skeleton image is encoded as a polygon. Therefore, each polygon element is taken from a fixed set of line elements (defined in Clause 6.2.4). The line starts at an offset coordinate with a starting direction and the following minutia characteristics:

— minutia type (2 bits: 00 virtual ending, 01 ridge ending, 10 ridge bifurcation, 11 virtual continuation);

— minutia direction (bit-depth defined in the record header, range: 0-360 degrees scaled according to bit-depth);

— x-coordinate (bit-depth defined in the record header);

— y-coordinate (bit-depth defined in the record header);

— number of direction elements following (8 bits).

The successive polygonal elements are defined by their direction change relative to the previous element or for the first element relative to the minutia direction, scaled and rounded to the direction code range and resolution (6.2.4). The length of each element is a function of the direction change (6.2.4):

— direction change (bit-depth and resolution defined in the record header, data type is a signed integer - the smallest negative number 10…0 is not used for direction change); (e.g. for bit-depth of 4 and 32 directions on $180°$ the signed integer range from –7 to 7 is scaled to the angle range from –39,375° to +39,375°;

  — or in situations of high ridge line curvature one may wish to store direction elements at higher spatial resolution. Therefore one can switch between two different resolution levels. With the smallest negative number 10…0 the resolution level is switched between normal or high. A line encoding will always start at normal resolution. On the first occurrence of 10…0 in a line code switch to high resolution level in using half the step length, on the second occurrence switch back normal resolution and full step length etc. (Table A.2).

— the direction change is repeated until the line end is reached;

— minutia type of line end (2 bits: 00 virtual ending, 01 ridge end, 10 ridge bifurcation, 11 virtual continuation).

If the skeleton line ends at a virtual ending (type number 00), the relative position of the minutia on the line element follows:

— The relative minutia position l/S_n is scaled to the range 0-3 via min(3, floor($4l/S_n$)) and stored as unsigned integer of length 2 bits, where l is the distance between the start of the last line element and the minutia, and S_n the step length of the last line element (Figure 2).

— If the skeleton line ends at a true minutia (type number 01 or 10) or is interrupted by a virtual continuation (type number 11) a byte-aligned minutia description follows. In order to keep the alignment overhead small it is done in the following manner: If the previously stored minutia type of the line end is already starting byte aligned, the minutia data is completed by appending its direction and position. On unaligned ending type, it is repeated at the start of the next byte followed by direction and position.

Thus the encoding continues with the following:

— if the previously stored minutia type of the line end is not starting byte aligned, it is repeated at the start of the next byte. Any unused bits caused by this alignment are filled with zeros;

— minutia direction (bit-depth defined in the record header, range 0-360 degrees scaled according to bit-depth);

— x-coordinate (bit-depth defined in the record header);

— y-coordinate (bit-depth defined in the record header).

If the ending minutia is of type virtual continuation (type number 11) the line description continues with

— the number of direction elements following (8 bits) and direction elements as described above.

Any unused bits of the last byte for each encoded line is filled with zeroes to get a byte aligned beginning for the next line encoding.

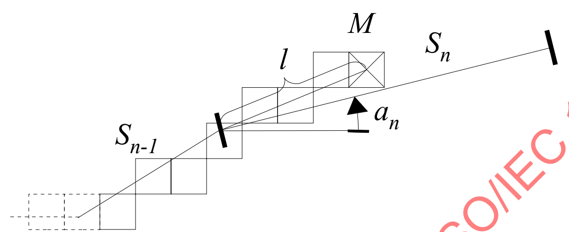


Figure 2 — The relative minutia position on a polygon line element is the ratio l/S_n, where S_n is the length of the line element passing the minutia M and l is the distance between the start of S_n and minutia M. α_n is the angle of S_n.

6.2.2 General skeleton line encoding rules

To keep the encoding size small a line shall start with a real minutia (type 01 or 10) if possible.

There are no restrictions about the use of virtual continuation minutiae or high resolution mode.

NOTE Virtual continuation minutia and the high resolution mode are “tools” to describe the ridges. One may prefer one method to describe high curvature and use the other to mark a line passing a bifurcation, a core or delta or extreme values in curvature. But these additional interpretations will increase the encoding size and can only be used in a non interoperable manner.

No assumption shall be made about the order of the line encodings in the record.

The skeleton shall be encoded only for image areas where the ridge lines are displayed with a sufficient quality (Figure A.3).

NOTE A one bit quality map is implicitly defined: At image areas with no encoded ridge line nearby the quality is 0 or not sufficient and at a image area with an encoded ridge line nearby the quality is 1 or sufficient. With the zonal quality data in the extended data area a multi-bit quality map may be defined in addition.

To judge the descriptive quality of the skeleton line encodings, one has to compare its reconstructed ridge lines with the fingerprint image the encoding comes from. The reconstructed ridge lines shall describe the fingerprint image in ridge position and structure, thus the following rules apply:

— The reconstructed skeleton line polygon element shall be inside the area of the ridge it is describing for most part of its length, i. e. at least 50%. A threshold in the range of 5% may be appropriate (best practice). This value depends on the reconstruction and comparison quality requirements of the application.

— The reconstructed skeleton line shall never be inside the area of any other ridge but the one it is describing.

— The reconstructed skeleton line shall preserve the topology of the ridges (see the definition of skeleton).

6.2.3 Constructing direction elements

For constructing the direction change α_i between two successive polygonal elements see Figure 3 and Figure 4. First, draw a circle, of radius equal to the polygon element length, around the current point. Obtain the intersection point between the circle and the skeleton line the in forward direction. The direction towards this point is scaled according to the bit-depth of the direction code. The difference between this direction and the previous line element is stored. The end point of this new polygon element with the fixed length and its digitised direction serves as the next starting point.

The previous construction is done with direction independent step size. For the general direction dependent step size replace the circle in the description above by the step size dependency defined in Clause 6.2.4.

In order to minimize integration of digitalisation error, each starting point must be computed with relatively high accuracy, i.e. its resolution shall be at least 100 times finer than the spatial resolution of the minutiae.

If the skeleton line ends during a step it is linearly extended to fill the polygon element length. The line encoding is completed with the minutiae type. For a true minutiae ending, its direction and the endpoint coordinates are stored. For a virtual ending, the relative minutia position on the current step is stored.

If the direction change of the skeleton line cannot be described by a direction element, the line encoding shall be interrupted by a “virtual continuation” and a new line encoding shall begin with the same point without repeating the minutia data.

A bifurcation (trifurcation, etc.) (Figure 4 and Figure A.2) is represented by two (or more) skeleton line encodings. One skeleton line passes the bifurcation without a real minutia at its position (Figure 4). All other lines end or start here and are assigned the type “bifurcation”. It is recommended to use the straightest ridge line passing the bifurcation without encoding a real minutia.

NOTE The most straight line is probably the dominant line, for which repetitive encodings with this part of ISO/IEC 19794 will not result in different line encodings - while the branching off line may swap from bifurcation to a ridge ending. I. e. depending from the sensor conditions in some images a bifurcation seems to be a ending with the dominant line passing through.

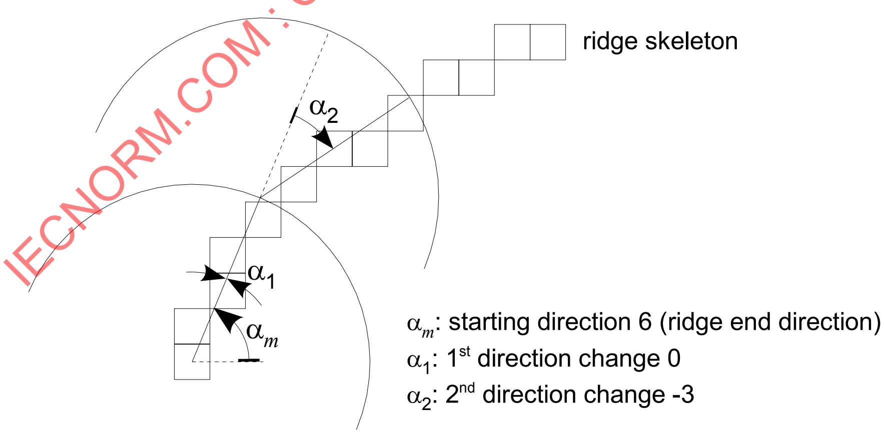


Figure 3 — The direction encoding starting from a skeleton end point. A bit-depth of 4 is used for direction change.

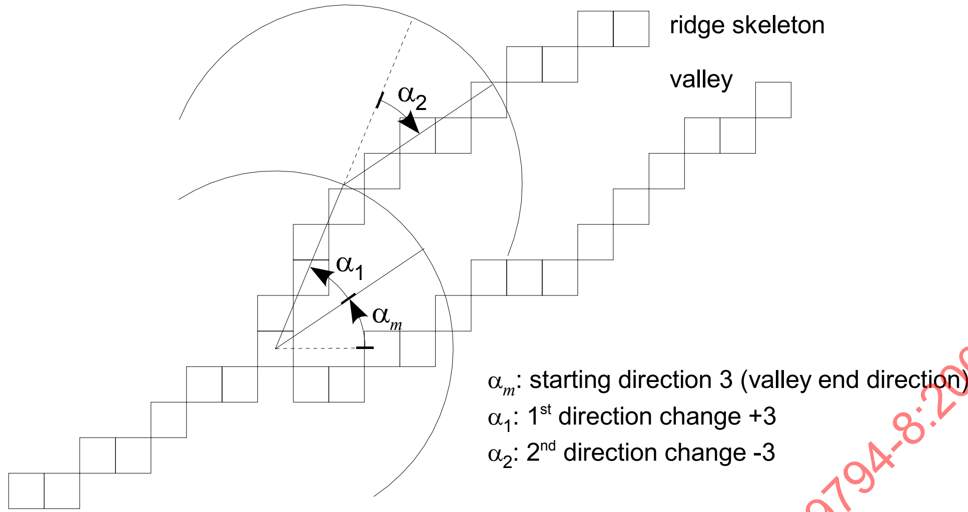


Figure 4 — The direction encoding starting from a skeleton bifurcation point. A bit-depth of 4 is used for direction change.

6.2.4 Direction element length

At most steps the direction change will be straight or nearly straight. With an increase step length on small direction changes and reduced angular range the number of direction elements are reduced.

The direction change dependant step size (Figure 5) and resolution is characterised by 4 parameters:

— The number of directions, N_π, on π or 180°. This gives the angular resolution, e. g. with N_π = 32 the resolution is 5,625°.

— With the bit-depth for one direction code element one gets the number of possible directions at each step. Since the change is symmetric to 0, the angular range is

$$\alpha_{max} = \pm(180° / N_\pi)\ (2^{\text{bit-depth} - 1} - 1) \quad (1)$$

NOTE With a resolution of 5,625° and at a bit-depth of 4, this gives a maximal bending of α_{max} = ±39,375°

— The step length for going straight, S_s.

— The maximal displacement perpendicular to the current direction, S_p. In the record header this value is stored relative to the straight step size, S_s, as 256 × S_p/S_s. If 256 × S_p/S_s is set to 0 in the record header a constant step length of S_s for all direction elements is used.

The design characteristics for the direction dependant step size are

— constant angular resolution, i. e. the distance between subsequent bending angles, α_n, is constant: $|\alpha_i - \alpha_{i\pm1}|$ = constant for all $i \in$ {.., -2, -1, 0, 1, 2, .. };

— constant spatial accuracy for all direction changes, i. e. the distance between subsequent steps, $\vec{r}_i$, is constant: $|\vec{r}_i - \vec{r}_{i\pm1}|$ = constant for all $i \in$ {.., -2, -1, 0, 1, 2, .. }.

With these conditions the endings of all of possible directions, $\vec{r}_i$, for one step are located on two circlular arches as shown in Figure 5. Thus the direction dependant step size, $|\vec{r}_i|$, is defined by

$$|\vec{r}_i| = \begin{cases} \dfrac{(S_s^{\,2} + 4S_p^{\,2})}{4S_p}\sin(2\varphi - |\alpha_i|) & \text{for } S_p > 0 \\ S_s & \text{for } S_p = 0 \end{cases} \qquad (2)$$

with the angle α_i between current direction and step $\vec{r}_i$ defined as

$$\alpha_i = 180° \, i / N_\pi \qquad (3)$$

and where

— $\varphi = \arctan(2S_p / S_s)$,

— $i \in \{.., -2, -1, 0, 1, 2, ..\}$ is the number of the direction change,

— S_s is the step length for going straight,

— S_p is the maximal displacement perpendicular to the current direction, and

— N_π is the number of directions on π or 180°.

An example for the angle dependant step size is given in Annex A.

NOTE The maximal curvature of the polygon is achieved with the minimal step size $r_{min} = r(\alpha_{max})$ from (2) at the maximal bending angle α_{max} from (1). A polygon with constant bending angle α_{max} and constant element length r_{min} has a radius R = 180° r_{min} / (π α_{max}). With S_s = 16, S_p = 3,75, and α_{max} = 39,375°, a minimal step length r_{min} = 3,9 and a radius of 5,7 pixel at a resolution of 100 ppcm is attained. At high resolution level the step length is cut in half, r_{min} = 1,95, thereby getting a radius of 2,85 pixels. With these settings a u-turn down to 0,6mm Sdiameter may be represented by a polygon without interruption by a virtual continuation minutia.

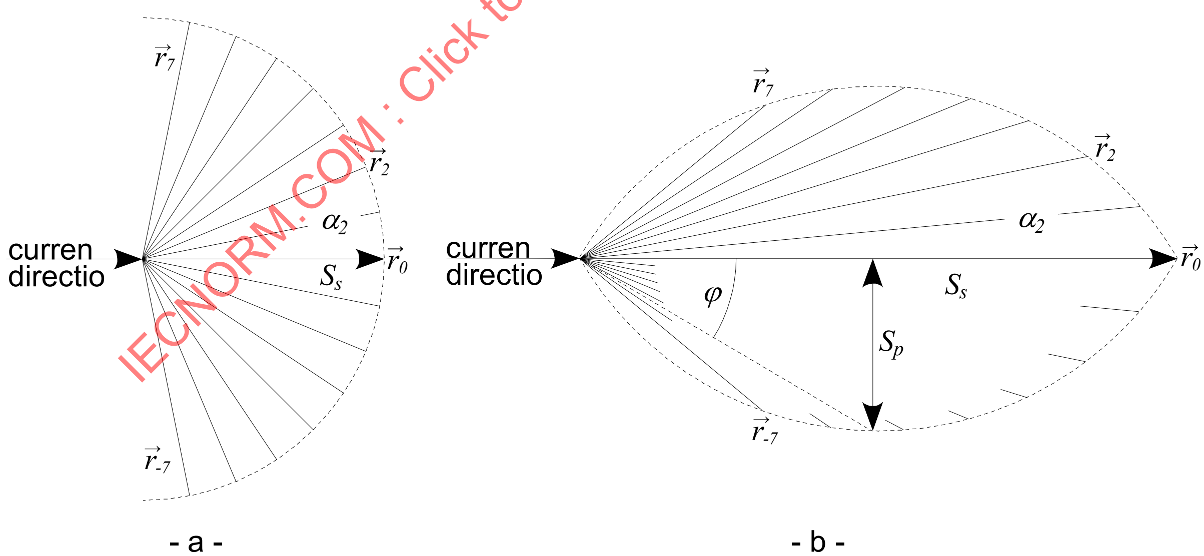


Figure 5 — Step length dependency of direction change:
(a) with S_p = 0 a constant step length is used, here at a angular resolution of 11,25°
(b) with S_p > 0 the steps at small bending angles are increased while steps at large bending angles are short. Here again 15 directions are encoded at higher angular example resolution of 5,625°.

6.3 Skeleton line neighbourhood index

The skeleton line gives the spatial connectivity in one dimension along the line. The direction perpendicular to the line is given by the neighbouring lines. Thus to help any comparison algorithm to analyse and compare a local two dimensional image area a link to the adjacent lines is very usefull. This link is given by list of neighbours for each encoded line (see A.4).

6.3.1 Adjacent lines

Two encoded ridge lines are neighbours to each other

a) if they are surrounding the same part of a valley

   1) for a not interupted distance of least the width of the valley,

   2) or for the whole line length of one of the lines (i.e. one of the lines is too short to comply with the condition 1),

b) and if the image has sufficient not interupted quality to support this ridge-valley-ridge-structure over the whole area needed to comply the condition a).

6.3.2 Recording the neighbour indices

The line index is the sequencial number of the encoded lines. A new line is starting with a starting minutia of any type (including continuation minutiae).

For each line with index number L one gets a list of neighbouring lines with indices A_i. If line 1 is a neighbour of line 2 also line 2 is a neighbour of line 1. So to get each neighbour relation only once, only lines with an index number $A_i \leq L$ are listed as neighbour of line L. This neighbourhood index list, including the line index L, is sorted by decreasing line index:

$L, A_1, \ldots, A_n$ where $L \geq A_1, A_1 > A_2, \ldots, A_{n-1} > A_n$,

where n is the number of neighbourhood entries for Line L. Since a line may be a neighbour to itself (e.g. at a u-turn), the first number in this list A_1 may be equal L. But since it is not usefull to list a neighourhood relation twice, any of the other indices shall be different i.e. $A_{i-1} > A_i$.

Then the subsequent differences between the line index L and the neighbour indices A_i are calculated:

$$L-A_1, A_1-A_2, \ldots, A_{n-1}-A_n, \quad (1)$$

The following data is recorded for one line:

— the number of neighbourhood entries for this line,

— followed by the list of index differences.

Concatenating the neighbour index data for all encoded lines in the same order as the line encodings in the record gives the skeleton line neighbouring index list.

The skeleton line neighbourhood index data starts with the bit-depth necessary to store the elements in the index list. The bit-depth is recorded in one byte followed by the neighbourhood index list, packed to bytes with a bit-depth given.

7 Finger pattern skeletal data record format

7.1 Introduction

The record format contains fields for both public and extended (proprietary) finger pattern skeletal interchange data. With the exception of the format identifier and the version number for the standard, which are null-terminated ASCII character strings, all data is represented in binary format. There are no record separators or field tags; fields are parsed by byte count.

7.2 Record organization

The organization of the record is as follows:

— a fixed-length (24-byte) record header containing information about the overall record, including the number of finger views represented and the overall record length in bytes;

— a single finger record for each finger, consisting of

  — a fixed-length (8-byte) header containing information about the data for a single finger;

  — the variable length fingerprint pattern skeletal description;

  — an extended data block containing the extended data block length and zero or more extended data areas for each finger.

All multi-byte quantities are represented in Big-Endian format; that is, the more significant bytes of any multi-byte quantity are stored at lower addresses in memory than (and are transmitted before) less significant bytes. Bit order follows the same endianness as the byte order. That is, the most significant bit is stored at the lowest bit address. All numeric values are fixed-length integer quantities.

7.3 Record header

There shall be one and only one record header for the finger pattern skeletal data record. The record header will contain information describing the identity and characteristics of the device that generated the data.

7.3.1 Format identifier

The finger pattern skeletal data record shall begin with a format identifier to be recorded in four bytes. For this part of ISO/IEC 19794, it shall consist of the three ASCII characters "FSK", followed by a zero byte as a NULL string terminator.

7.3.2 Version number

The version number for the version of this part of ISO/IEC 19794 used in constructing the record shall be placed in four bytes. This version number shall consist of three ASCII numerals followed by a zero byte as a NULL string terminator. The first and second character will represent the major revision number and the third character will represent the minor revision number. Upon approval of this specification, the version number shall be "010" (an ASCII '0'followed by an ASCII '1' and an ASCII '0').

7.3.3 Length of record

The length of the entire record shall be recorded in four bytes.

7.3.4 Capture equipment certifications

This field contains four bits used to indicate if the capture equipment used to capture the original fingerprint image was compliant with a standard certification method for such equipment. Currently, only two bits are defined. If the most significant bit is '1', the original capture equipment was certified to be compliant with some national body capture equipment certification. The least significant of the four bits is reserved for a future ISO finger image capture equipment certification. The two additional bits are reserved for future image quality certifications.

7.3.5 Capture device ID

The capture device type ID shall be recorded in twelve bits. This ID is used to identify the type or model of the capture device used to acquire the original biometric sample. A value of all zeros will be acceptable and will indicate that the capture device type ID is unreported. The vendor determines the value for this field, if not regulated otherwise in an application context. Applications developers and users may obtain the values for these codes, as well as the model(s) corresponding to a particular ID, from the vendor. Reporting the capture device type ID is optional but recommended. The value "unreported" may not be allowable in some applications.

7.3.6 Number of finger views in record

The total number of finger views represented in the record shall be contained in 1 byte.

7.3.7 Resolution of scaled image

The resolution (in ppcm) of the scaled finger image(s) shall be uniform in the x and y-directions and shall be stored in 1 byte.

7.3.8 Bit-depth of direction code start and stop point coordinates

The bit-depth used to represent the x and y-coordinate of the starting and ending point in the direction code description of the skeleton shall be recorded in 1 byte.

7.3.9 Bit-depth of direction code start and stop direction

The bit-depth used to represent the direction of the starting and ending point in the direction code description of the skeleton shall be recorded in 1 byte.

7.3.10 Bit-depth of direction in direction code

The bit-depth used to represent the direction in the direction code shall be recorded in 1 byte.

7.3.11 Step size of direction code

The maximal step size S_s in the current direction of each direction code step shall be recorded in 1 byte.

7.3.12 Relative perpendicular step size of direction code

The relative perpendicular step size floor(256 $\times$ S_p/S_s) of the direction code shall be recorded in 1 byte.

7.3.13 Number of directions on 180°

The angular resolution of the direction code is stored as the number N_π of directions on 180° and shall be recorded in 1 byte.

7.3.14 Reserved bytes

Two bytes are reserved for future revision of this specification. For version 1.0 of this part of ISO/IEC 19794, these byte values shall be set to 0.

7.4 Single finger record format

7.4.1 Finger header

A finger header shall start each area of finger data providing information for that finger. There shall be one finger header for each finger contained in the finger pattern skeletal data record. The finger header will occupy a total of 10 bytes as described below. Note that it is permissible for more than one single finger record to represent the same finger, with (presumably) different data.

7.4.1.1 View number

If more than one finger pattern record in a general record is from the same finger, each pattern record shall have a unique view number. The combination of finger location and view number shall uniquely identify a particular pattern record within a general record. Multiple finger pattern records from the same finger shall be numbered with increasing view numbers, beginning with zero. Where only one finger pattern record is taken from each finger, this field shall be set to 0. The view number shall be recorded in one byte.

7.4.1.2 Finger position

The finger position shall be recorded in one byte. The codes for this byte shall be as defined in Table 5 of ANSI/NIST-ITL 1-2000, “Data format for the interchange of fingerprint information”. This table is reproduced here in Table 1 for convenience. Only codes 0 through 10 shall be used, the “plain” codes included in Table 5 of ANSI/NIST ITL 1-2000 are not relevant for this part of ISO/IEC 19794.

Table 1 — Finger position codes

Finger position	Code
Unknown finger	0
Right thumb	1
Right index finger	2
Right middle finger	3
Right ring finger	4
Right little finger	5
Left thumb	6
Left index finger	7
Left middle finger	8
Left ring finger	9
Left little finger	10

7.4.1.3 Impression type

The impression type of the finger images that the finger pattern skeletal data was derived from shall be recorded in one byte. The codes for this byte are shown in Table 2. These codes are derived from Table 4 of ANSI/NIST-ITL 1-2000, “Data format for the interchange of fingerprint information”, with the addition of the “swipe” type. The “swipe” type identifies data records derived from image streams generated by sliding the finger across a small sensor. Only codes 0 through 3 and 8 through 9 shall be used; the “latent” codes are not relevant for this part of ISO/IEC 19794.

Table 2 — Impression type codes

Description	Code
Live-scan plain	0
Live-scan rolled	1
Nonlive-scan plain	2
Nonlive-scan rolled	3
Latent impression	*4*
Latent tracing	*5*
Latent photo	*6*
Latent lift	*7*
Swipe	8
Live-scan contactless	9

7.4.1.4 Finger quality

The quality of the overall finger pattern skeletal data shall be between 0 and 100 and recorded in one byte. This quality number is an overall expression of the quality of the finger record, and represents quality of the original image, of the pattern extraction and any additional operations that may affect the pattern record. A value of 0 shall represent the lowest possible quality and the value 100 shall represent the higher possible quality. The numeric values in this field shall be set in accordance with the general guidelines contained in of ISO/IEC 19784-1:2006, *Information technology – Biometric application programme interface – BioAPI specification*. The comparison algorithm may use this value to determine its certainty of verification.

7.4.1.5 Size of skeleton image in x direction

The size of the skeleton image in pixels in the x direction shall be contained in two bytes.

7.4.1.6 Size of skeleton image in y direction

The size of the skeleton image in pixels in the y direction shall be contained in two bytes.

7.4.1.7 Length of finger pattern skeletal data block

The length (in bytes) of the finger pattern skeletal data block recorded for the finger shall be recorded in two bytes. The length provided includes any padding bits necessary to complete the last byte of finger pattern skeletal data.

7.4.2 Finger pattern skeletal data block

The finger pattern skeletal data block for a single finger has two parts: the finger pattern skeletal data and the skeleton line neighbourhood index data. Each part is recorded together with a length descriptor as follows.

7.4.2.1 Length of finger pattern skeletal data

The length (in bytes) of the finger pattern skeletal data shall be recorded in two bytes. The length provided includes any padding bits necessary to complete the last byte of finger pattern skeletal data.

7.4.2.2 Finger pattern skeletal data

The finger pattern skeletal data for a single finger shall be recorded as defined in Clauses 6.1 and 6.2.

7.4.2.3 Length of skeleton line neighbourhood index data

The length (in bytes) of the skeleton line neighbourhood index data shall be recorded in two bytes. The length provided includes any padding bits necessary to complete the last byte of finger pattern skeletal data.

7.4.2.4 Skeleton line neighbourhood index data

The skeleton line neighbourhood index data for a single finger shall be recorded as defined in Clause 6.3.

7.5 Extended data

The extended data area of the finger pattern skeletal data record is open to placing additional data that may be used by the comparison equipment. The size of this area shall be kept as small as possible, augmenting the data stored in the standard pattern skeletal data area. The extended data for each finger view shall immediately follow the standard pattern skeletal data for that finger view and shall begin with the extended data block length. More than one extended data area may be present for each finger and the extended data block length will be the summation of the lengths of each extended data area. The data block length is used as a signal for the existence of the extended data while the individual extended data length fields are used as indices to parse the extended data. Note that the extended data area cannot be used alone, without the standard portion of the pattern skeletal data record.

While the extended data area allows for inclusion of proprietary data within the pattern skeletal format, this is not intended to allow for alternate representations of data that can be represented in open manner as defined in this part of ISO/IEC 19794. In particular, ridge count data, core and delta data, zonal quality information or sweat pore positions shall not be represented in proprietary manner to the exclusion of the publicly defined formats in this part of ISO/IEC 19794. Additional ridge count, core and delta, zonal quality information or sweat pore positions may be placed in a proprietary extended data area if the standard fields defined below are also populated. The intention of this part of ISO/IEC 19794 is to provide interoperability.

7.5.1 Common extended data fields

7.5.1.1 Extended data block length

All pattern skeletal data records shall contain the extended data block length. This field will signify the existence of extended data, and shall be recorded in 2 bytes. A value of all zeros (0x0000 hexadecimal) will indicate that there is no extended data and that the file will end or continue with the next finger view. A nonzero value will indicate the length of all extended data starting with the next byte.

7.5.1.2 Extended data area type code

The extended data area type code shall be recorded in two bytes, and shall distinguish the format of the extended data area as defined by the Vendor specified by the CBEFF_BDB_product_owner and CBEFF_BDB_product_type in the CBEFF header. A value of zero in both bytes is a reserved value and shall not be used. A value of zero in the first byte, followed by a non-zero value in the second byte, shall indicate that the extended data area has a format defined in this part of ISO/IEC 19794. A non-zero value in the first byte shall indicate a vendor specified format, with a code maintained by the vendor. Refer to Table 3 for a summary of the extended data area type codes. If the extended data block length (7.5.1.1) for the finger view is zero, indicating no extended data, this field shall not be present.

NOTE If vendor defined extended data is present and the Standard Biometric Header (SBH) does not support CBEFF_BDB_product_owner and CBEFF_BDB_product_type, then the link between the extended data and the vendor will be lost.

Table 3 — Extended data area type codes

First byte	Second byte	Identification
0x00	0x00	reserved
0x00	0x01	ridge count data (Clause 7.5.2)
0x00	0x02	core and delta data (Clause 7.5.3)
0x00	0x03	zonal quality data (Clause 7.5.4)
0x00	0x04	sweat pore position data (Clause 7.5.5)
0x00	0x05	skeleton structural data (Clause 7.5.6)
0x00	0x06-0xFF	reserved
0x01-0xFF	0x00	reserved
0x01-0xFF	0x01-0xFF	vendor-defined extended data

7.5.1.3 Extended data area length

The length of the extended data area, including the extended data area type code and length of data fields, shall be recorded in two bytes. This value is used to skip to the next extended data if the comparison algorithm cannot decode and use this data. If the extended data block length (7.5.1.1) for the finger view is zero, indicating no extended data, this field shall not be present.

7.5.1.4 Extended data area

The extended data area field of the extended data block is defined by the equipment that is generating the finger pattern skeletal data record, or by common extended data formats contained in this part of ISO/IEC 19794; see Clauses 7.5.2, 7.5.3, 7.5.4, 7.5.5 and 7.5.6. If the extended data block length (7.5.1.1) for the finger view is zero, indicating no extended data, this field shall not be present.

7.5.2 Ridge count data format

If the extended data area type code is 0x0001, the extended data area contains ridge count information. This format is provided to contain optional information about the number of fingerprint ridges between pairs of minutiae points. Each ridge count is associated with a pair of minutiae points contained in the finger pattern skeletal data area defined in Clause 7.4.2; no ridge information may be contained that is associated with minutiae not included in the corresponding skeletal data area. Ridge counts shall not include the ridges represented by either of the associated minutiae points. Refer to Figure 6 for clarification; the ridge count between minutiae A and B is 1, while the ridge count between minutiae B and C is 2.

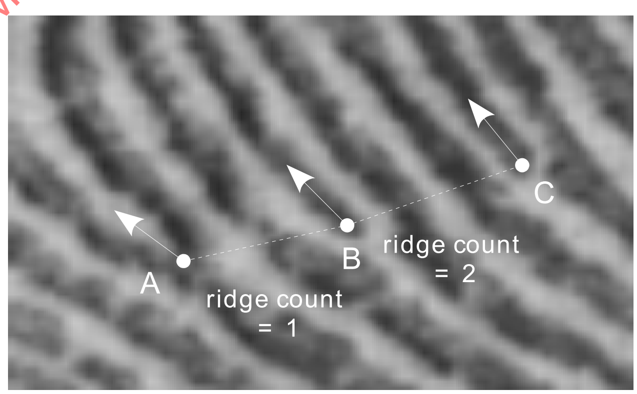


Figure 6 — Example ridge count data. Note the difference of the minutiae positions to ISO/IEC 19794-2 as mentioned in 6.1.4

7.5.2.1 Ridge count extraction method

The ridge count data area shall begin with a single byte indicating the ridge count extraction method. Ridge counts associated with a particular centre minutiae point are frequently extracted in one of two ways: by extracting the ridge count to the nearest neighbouring minutiae in each of four angular regions (or quadrants), or by extracting the ridge count to the nearest neighbouring minutiae in each of eight angular regions (or octants). The ridge count extraction method field shall indicate the extraction method used, as shown in Table 4.

Table 4 — Ridge count extraction method codes

RCE method field value	Extraction method	Comments
0x00	Non-specific	No assumption shall be made about the method used to extract ridge counts, nor their order in the record; in particular, the counts may not be between nearest-neighbour minutiae
0x01	Four-neighbour (quadrants)	For each centre minutiae used, ridge count data was extracted to the nearest neighbouring minutiae in four quadrants, and ridge counts for each centre minutiae are listed together
0x02	Eight-neighbour (octants)	For each centre minutiae used, ridge count data was extracted to the nearest neighbouring minutiae in eight octants, and ridge counts for each centre minutiae are listed together

If either of these specific extraction methods are used, the ridge counts shall be listed in the following way:

— all ridge counts for a particular centre minutiae point shall be listed together;

— the centre minutiae point shall be the first minutiae point references in the three-byte ridge count data;

— if a given quadrant or octant has no neighbouring minutiae in it, a ridge count field shall be recorded with both the minutiae index and the ridge count fields set to zero (so that, for each centre minutiae, there shall always be four ridge counts recorded for the quadrant method and eight ridge counts recorded for the octant method);

— no assumption shall be made regarding the order of the neighbouring minutiae.

7.5.2.2 Ridge count data

The ridge count data shall be represented by a list of three-byte elements. The first and second bytes are an index number, indicating which minutiae points in the corresponding finger pattern skeletal data area are being considered. The minutia index is the number of occurrences of a real minutia encoded in the skeletal data area. The index begins at 1. The third byte is a count of the ridges intersected by a direct line between these two minutiae points.

There is no requirement that the ridge counts be listed with the lowest index number first. Since the minutiae points are not listed in any specified geometric order, no assumption shall be made about the geometric relationships of the various ridge count items.

7.5.2.3 Ridge count format summary

The ridge count data format shall be as follows:

7.5.2.1 Extraction type	7.5.2.2 index #1	7.5.2.2 index #2	7.5.2.2 ridge count		7.5.2.2 index #1	7.5.2.2 index #2	7.5.2.2 ridge count
Method	*index*	*Index*	*count*	…	*index*	*index*	*count*
1 byte	1 byte	1 byte	1 byte		1 byte	1 byte	1 byte
	3 bytes				3 bytes		

7.5.3 Core and delta data format

If the extended data area type code is 0x0002, the extended data area contains core and delta information. This format is provided to contain optional information about the placement and characteristics of the cores and deltas on the original fingerprint image. Core and delta points are determined by the overall pattern of ridges in the fingerprint. There may be zero or more core points and zero or more delta points for any fingerprint. Core and delta points may or may not include angular information.

The core and delta information shall be represented as follows. The first byte shall contain the number of core points included; valid values are 0 to 15.

7.5.3.1 Number of cores

The number of core points represented shall be recorded in the least significant four bits of this byte. Valid values are from 0 to 15. The most significant four bits of this byte shall be reserved for future revision of this specification. For version 1.0 of this part of ISO/IEC 19794, these bit values shall be set to 0.

7.5.3.2 Core information type

The core information type shall be recorded in the two most significant bits of the two bytes of the x coordinate of the core position. The bits "01" will indicate that the core has angular information while "00" will indicate that no angular information is relevant for the core type. If this field is "00", then the angle field shall not be present for this core.

7.5.3.3 Core position

If there are ridge endings enclosed by the innermost recurving ridgeline, the ending nearest to the maximal curvature of the recurving ridgeline defines the core position. If the core is a u-turn of a ridgeline not enclosing ridge endings, the valley end defines the core position.

The x coordinate of the core shall be recorded in the least significant fourteen bits of the first two bytes (fourteen bits). The y coordinate shall be placed in the least significant fourteen bits of the following two bytes. The most significant two bits of these two bytes shall be reserved for future revision of this specification. For version 1.0 of this part of ISO/IEC 19794, these bit values shall be set to 0. The coordinates shall be expressed in pixels at the resolution indicated in the record header.

7.5.3.4 Core angle

If the core has a discernible angle of direction it shall be recorded in the core information, since this characterises the type of core. The core has a direction if there is a ridge or a group of ridges pointing towards it. The angle of a core is defined by the angle of the tangent to these ridge lines as close as possible to the core position. The tangent is pointing to the open side of the U-structured ridge.

The angle of the core shall be recorded in one byte in units of 1,40625 (360/256) degrees. The core angle is measured increasing counter-clockwise starting from the horizontal axis to the right. The value shall be a non-negative value between 0 and 255, inclusive. For example, an angle value of 16 represents 22,5 degrees. If the core information type is zero (see Clause 7.5.3.2), then this field shall not be present for this core.

7.5.3.5 Number of deltas

The number of delta points represented shall be recorded in the least significant four bits of this byte. Valid values are from 0 to 15. The most significant four bits of this byte shall be reserved for future revision of this specification. For version 1.0 of this part of ISO/IEC 19794, these bit values shall be set to 0.

7.5.3.6 Delta information type

The delta information type shall be recorded in the two most significant bits of the two bytes of the x coordinate of the delta position. The bits “01” will indicate that the delta has angular information while “00” will indicate that no angular information is relevant for the delta type. If this field is “00”, then the angle fields shall not be present for this delta.

7.5.3.7 Delta position

For a delta there are three points of divergences each placed between the two ridges at the location where the ridges begin to diverge; that is, where the ridges that have been parallel or nearly parallel begin to spread apart as they approach the delta. The position of the delta is defined by the spatial mean of these three points.

The x coordinate of the delta shall be recorded in the least significant fourteen bits of the first two bytes (fourteen bits). The y coordinate shall be placed in the least significant fourteen bits of the following two bytes. The most significant two bits of these two bytes shall be reserved for future revision of this specification. For version 1.0 of this part of ISO/IEC 19794, these bit values shall be set to 0. The coordinates shall be expressed in pixels at the resolution indicated in the record header.

7.5.3.8 Delta angles

For all observable divergences the angle is defined by the direction of the tangent before the pair of ridges begins to diverge. The angle shall point from divergent towards parallel lines; that is, the angles shall point outwards from the delta.

The three angle attributes of the delta shall each be recorded in one byte in units of 1,40625 (360/256) degrees. The delta angle is measured increasing counter-clockwise starting from the horizontal axis to the right. The value shall be a non-negative value between 0 and 255, inclusive. For example, an angle value of 16 represents 22,5 degrees. If the delta information type is zero (see Clause 7.5.3.6), then this field shall not be present. If not all three angles can be extracted from the image because of noise or image cropping, the angle fields affected shall be filled by repeating any of the other angle(s) for the same delta.

7.5.3.9 Core and delta format summary

The core format shall be as follows:

	7.5.3.1	7.5.3.2	7.5.3.3		7.5.3.3	7.5.3.4	7.5.3.2, 7.5.3.3 & 7.5.3.4
Reserved	# of cores	Core info type	X location	Reserved	Y location	Core Angle	…
Reserved	*# cores*	*Type*	*x coordinate*	*reserved*	*y coordinate*	*angle*	…
4 bits	4 bits	2 bits	14 bits	2 bits	14 bits	1 byte	5 bytes
1 byte		2 bytes		2 bytes		only present if core info type not zero	zero or more additional cores

The delta format shall be as follows:

	7.5.3.5	7.5.3.6	7.5.3.7		7.5.3.7	7.5.3.8	7.5.3.6, 7.5.3.7 & 7.5.3.8
Reserved	# of deltas	Delta info type	X location	Reserved	Y location	Delta Angles	… … …
Reserved	*# deltas*	*Type*	*x coordinate*	*reserved*	*y coordinate*	*ang1* | *ang2* | *ang3*	… | … | …
4 bits	4 bits	2 bits	14 bits	2 bits	14 bits	3 bytes	7 bytes
1 byte		2 bytes		2 bytes		only present if delta info type not zero	zero or more additional deltas

7.5.4 Zonal quality data

If the extended data area type code is 0x0003, the extended data area contains zonal quality data. This format is provided to contain optional information about the quality of the fingerprint image within each cell in a grid defined on the original fingerprint image. Within each cell, the quality may depend on the presence and clarity of ridges, spatial distortions and other characteristics.

The zonal quality data shall be represented as follows. The first three bytes shall contain the horizontal, the vertical cell sizes in pixels and the bit-depth of the cell quality information. These size bytes shall be followed by the quality indications for each cell. All cells are the same size, with the exception of the final cells in each row and in each column. The final cell in each row and in each column may be less than the stated cell size, if the cell width and height are not factors of the image width and height, respectively.

7.5.4.1 Cell width and height

The number of pixels in cells in the x-direction (horizontal) shall be stored in one byte. Permissible values are 1 to 255. The number of pixels in cells in the y-direction (vertical) shall be stored in one byte. Permissible values are 1 to 255.

7.5.4.2 Cell quality information depth

The bit-depth of the cell quality information shall be contained in one byte. This value will indicate the number of bits per cell used to indicate the quality.

7.5.4.3 Cell quality data

The quality of the fingerprint image in each cell shall be represented by one or more bits, as indicated in 7.5.4.2. Quality data for cells shall be stored in usual “raster” order – left to right, then top to bottom. If the finger image within this cell is of good clarity and significant ridge data is present, the cell quality shall be represented by higher values (by the bit value ‘1’ if the information depth is 1). If the cell does not contain significant ridge data, or the ridge pattern within the cell is blurred, broken or otherwise of poor quality, the cell quality shall be represented by lower values (the bit value ‘0’ if the information depth is 1).

The cell quality shall be packed into bytes. The final byte in the cell quality data may be packed with bit values of zero (‘0’) for the least significant bits as required to complete the last byte.

7.5.4.4 Zonal quality data format summary

The zonal quality data format shall be as follows:

7.5.4.1 Cell Width	7.5.4.1 Cell Height	7.5.4.2 Information Depth	7.5.4.3 Cell Quality Data	
x cell size	*y cell size*	*depth*	*Cell quality bits*	*00…0*
1 byte	1 byte	1 byte	data bits	padding bits

7.5.5 Sweat pore position data

The position, size and shape of sweat pores are unique characteristic features that can enhance pattern and minutiae based verification. A fingerprint image with clearly visible pores is depicted in Figure 7. A fingerprint image may contain as many as 2700 sweat pores. Their size shape and location can be used as features for fingerprint comparison. In this part of ISO/IEC 19794 only the sweat pore position along the skeleton line is encoded. [7]

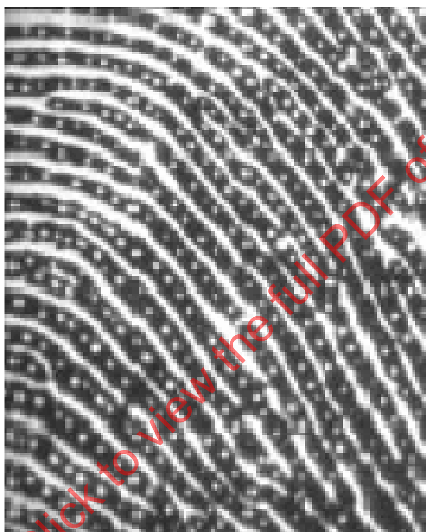


Figure 7 — Fingerprint fragment with sweat pores (image from [6]).

The encoding of the sweat pore positions starts with a 3 byte header containing the resolution (2 bytes) and the bit-depth of each description element. Then for each skeleton line in 7.4.2 a series of sweat pore distance description elements follows.

7.5.5.1 Sweat pore position resolution

The resolution of the sweat pore position description is stored in 2 bytes. The minimum recommended value is 200 ppcm.

7.5.5.2 Sweat pore distance information depth

The bit-depth of the sweat pore distance elements is stored in one byte. The valid range is 2 to 8 bits, minimum recommended is 4.

7.5.5.3 Sweat pore position description

For each skeleton line in 7.4.2 the sweat pore distance description starts with the value 00…0. A series of sweat pore distance elements describe the position of the sweat pores along the skeleton line. Values 00…1 to 11…0 give the successive distance between sweat pores. The value 11…1 indicates that there is no sweat pore up to the distance 11…0. Thus each element of bit-depth in 7.5.5.2 has a valid distance range from 1 to ($2^{bitdepth}$–2) at the resolution in 7.5.5.1. The final byte in the sweat pore position data shall be packed with bit values of zero (‘0’) for the least significant bits as required to complete the last byte.

7.5.5.4 Sweat pore position format summary

The sweat pore position data format shall be as follows:

7.5.5.1 Position reslousion	7.5.5.2 Information depth	7.5.5.3 Sweat pore position description	
Resolution [ppcm]	*Depth*	*Distance elements*	*00…0*
2 bytes	1 byte	data bits	padding bits

7.5.6 Finger pattern skeleton structural data

For some comparison algorithms it may be useful to follow all ridge lines ending at a real minutia. With the skeleton description to get the ridge near the ending minutiae the whole line has to be reconstructed. To find all ridges starting from a bifurcation the situation is even worse, an extensive search has to be done to find the line passing the bifurcation.

To reconstruct a ridge line in reversed order one needs the direction of the last direction element, the step length of the last step and the resolution level (6.2.1).

To know the line passing a bifurcation, the line number is needed. To reconstruct this line starting at the bifurcation the exact position of the minutia on this line, the direction of the corresponding element and the resolution level (6.2.1) have to be provided.

The skeleton structural data starts with the bit-depth necessary to store the line index. This bit-depth is stored in one byte and has the range from 4 to 16. The structural information following has the same order as the real minutiae in the skeleton data and it is stored in a data packed compacted bit form with no record separators or field tags.

For real minutiae at the end of a skeleton line the following data is stored.

— Type of the structural data element, here 0 for line end information. This value is stored with a bit-depth of 1.

— Direction of the last polygon line element (α_n in Figure 2) with the same angular resolution as the direction elements in direction code (N_π/π) and a bit-depth necessary to store $2N_\pi$–1.

— Relative position of the minutia on line element min(S_s–1, floor($S_s l/S_n$)), where l is the distance between the start of the last line element and the minutia and S_n the step length of the last line element, see Figure 2. This value is stored with a bit-depth necessary to store S_s–1.

— Resolution level with the values 0 for normal 1 for high resolution. This value isstored with a bit-depth of 1.

For each bifurcation store the following.

— Type of the structural data element, here 1 for information about a bifurcation. This value is stored with a bit-depth of 1.

— Line number, starting with index origin 0. This value is stored with a bit of 8.

— Line element number, starting with index origin 0. The bit-depth for this value is defined in the first byte of the skeleton structural data.

— Direction of the polygon line element passing the bifurcation (α_n in Figure 2) with the same angular resolution as the direction elements in direction code (N_π/π) and a bit-depth necessary to store $2N_\pi-1$.

— Relative position of the bifurcation on line element min(S_s–1, floor($S_s l/S_n$)), where l is the distance between the start of the last line element passing the bifurcation and the minutia and S_n the step length of the last line element, see Figure 2. This value is stored with a bit-depth necessary to store S_s–1.

— Resolution level with the values 0 for normal 1 for high resolution. This value is stored with a bit-depth of 1.

Where S_s is the step length for going straight and N_π is the number of directions on π or 180° (6.2.4), both defined in the record header.

A minutia of type 'ridge ending' encoded at the beginning of a skeleton line has no entry in the skeleton structural data. For a minutia of type 'ridge ending' encoded at the end of the skeleton line the line end information is stored (type 0). For a minutia of type 'ridge bifurcation' encoded at the beginning of a skeleton line the bifurcation information (type 1) is stored. For a minutia of type 'ridge bifurcation' encoded at the end of a skeleton line, first store the line end information (type 0) followed by the bifurcation information (type 1).

7.5.6.1 Finger pattern skeleton structural data format summary

The finger pattern skeleton structural data format shall be as follows:

Index depth	…	Type line end	Direction	Realtive position	Resolution level	…
depth N_{idx}	…	*0*	*direction*	*position*	*level*	…
1 byte	…	1 bit	N_d bits	N_p bits	1 bit	…
	…	For each real minutiae at the end of a skeleton line				…

…	Type bifurcation	Line number	Element number	Direction	Relative position	Resolution level	…
…	*1*	*number*	*Number*	*direction*	*position*	*level*	…
…	1 bit	8 bits	N_{idx} bits	N_d bits	N_p bits	1 bit	…
…			For each bifurcation				…

…	
…	*00…0*
…	padding bits
…	

where

— N_d is the bit-depth necessary to store $2N_\pi-1$, e.g. with N_π=32 gives N_d = 6 bit.

— N_p is the bit-depth necessary to store S_s–1, e.g. with S_s = 16 gives N_p = 4 bit.

— N_{idx} is the bit-depth necessary to store the line index. This number is given with the first byte of the structural data.

7.6 Pattern record format summary

Table 5 is a reference for the fields present in the finger pattern skeletal data record format. Optional extended data formats for ridge counts, core and delta data, zonal quality information and sweat pore position data are not represented here.

Table 5 — Pattern record format summary

	Field	Size	Values	Notes
One per Record	Format Identifier	4 bytes	0x46534b00 ('FSK' 0x0)	"FSK " – finger pattern skeletal record
	Version of this part of ISO/IEC 19794	4 bytes	N n n 0x0	" XX"
	Length of total record in bytes	4 bytes		
	Capture equipment certification	4 bits		Compliance with NB or future ISO standards
	Capture device ID	12 bits		Vendor specified
	Number of finger views in record	1 byte	1-255	
	Resolution of finger pattern [ppcm]	1 byte	1-255	Recommended 100ppcm
	Bit-depth of direction code start and stop point coordinates	1 byte	8-16	Recommended 8
	Bit-depth of direction code start and stop direction	1 byte	4-8	Recommended 6
	Bit-depth of direction in direction code	1 byte	3-8	Recommended 4
	Step size of direction code S_s	1 byte	1-255	Recommended 16
	Relative perpendicular step size $256 \times S_p/S_s$	1 byte	0-255	Recommended 60
	Number N_π of directions on 180°	1 byte	1-255	Recommended 32
	Reserved bytes	2 bytes	00	RFU
One per View	View number	1 byte	0 to 15	
	Finger position	1 byte	0 to 11	Refer to ANSI/NIST standard (Table 1)
	Impression type	1 byte	0 to 3 or 8	(Table 2)
	Finger quality	1 byte	0 to 100	0 to 100
	Skeleton image size in x	2 bytes		in pixels
	Skeleton image size in y	2 bytes		in pixels
	Length of finger pattern skeletal data block	2 bytes		
	Length of finger pattern skeletal data	2 bytes		
	Finger pattern skeletal data	In prev. field		
	Length of skeleton line neighbourhood index data	2 bytes		
	Skeleton line neighbourhood index data	In prev. field		
	Extended data block length	2 bytes		0x0000 = no extended area
0+ per View	Extended data area type code	2 bytes		only present if extended data block length ≠ 0
	Extended data area length	2 bytes		*Each extended data area may contain vendor-specific data, or one or more of the following (in any order):*
	Extended data	In prev. field		— *Ridge count data,* — *Core and delta data* — *Zone quality data* — *Sweat pore position data* — *Skeleton structural data*

8 Finger pattern skeletal data card format

This part of ISO/IEC 19794 defines two card related encoding formats for finger pattern skeletal data, the normal size format and the compact size format. Such a format may be used e.g. as part of a Biometric Information Template as specified in ISO/IEC 7816-11 with incorporated CBEFF data objects, if off-card comparison is applied, or in the command data field of a VERIFY command, if comparison-on-card (CoC) is applied (see ISO/IEC 7816-4 and -11).

The two card formats represent two sets of fixed parameters (see 8.1 and 8.2). These fixed values are not included in the card format.

NOTE For the record format these pararmeters are set in the record header (7.3).

The finger pattern skeletal data card format consists of the finger pattern skeletal data block as defined in Clause 8.3 and optionally additional features (see Clause 8.5).

NOTE The term “card” is used for smartcards as well as for other kind of tokens.

8.1 Normal size finger pattern skeletal format

For the normal size format most of the header entries for the record format get fixed to the following values:

— Resolution of direction code start and stop point	200 ppcm
— Bit-depth of direction code start and stop point in x	11
— Bit-depth of direction code start and stop point in y	11
— Bit-depth of direction code start and stop direction	8
— Bit-depth of direction in direction code	4
— Step size S_s of direction code	24
— Relative perpendicular step size $256 \times S_p/S_s$	60
— Number N_π of directions on 180°	32

8.2 Compact size finger pattern skeletal format

For the compact size format most of the header entries for the record format get fixed to the following values:

— Resolution of direction code start and stop point	100 ppcm
— Bit-depth of direction code start and stop point in x	8
— Bit-depth of direction code start and stop point in y	8
— Bit-depth of direction code start and stop direction	6
— Bit-depth of direction in direction code	4
— Step size S_s of direction code	16
— Relative perpendicular step size $256 \times S_p/S_s$	60
— Number N_π of directions on 180°	32

8.3 Finger pattern skeletal data block

The finger pattern skeletal data block for a single finger has two parts: the finger pattern skeletal data and the skeleton line neighbourhood index data. Each part is recorded together with a length descriptor as follows.

8.3.1 Skeleton image size in x and y

The skeleton image size in pixels in x is stored in 2 bytes at a resolution of 100 ppcm for the compact and 200 ppcm for the normal format.

The skeleton image size in pixels in y is stored in 2 bytes at a resolution of 100 ppcm for the compact and 200 ppcm for the normal format.

8.3.2 Length of finger pattern skeletal data

The length (in bytes) of the finger pattern skeletal data shall be recorded in two bytes. The length provided includes any padding bits necessary to complete the last byte of finger pattern skeletal data.

8.3.3 Finger pattern skeletal data

The finger pattern skeletal data for a single finger shall be encoded as defined in Clauses 6.1 and 6.2 with the definitions of 8.1 respectively. 8.2. If no sorting of the skeleton lines is necessary according to 8.4 their sequence is arbitrary.

8.3.4 Length of skeleton line neighbourhood index data

The length (in bytes) of the skeleton line neighbourhood index data shall be recorded in two bytes. The length provided includes any padding bits necessary to complete the last byte of finger pattern skeletal data.

8.3.5 Skeleton line neighbourhood index data

The skeleton line neighbourhood index data for a single finger shall be recorded as defined in Clause 6.3.

8.4 The x or y coordinate extension for compact card format

If the x value of the skeleton image size (8.3.1) is greater than 255, the direction code must be sorted. The sorting will be performed according to the ascending x-coordinate of the starting position of the direction code.

If the y value of the skeleton image size (8.3.1) is greater than 255, the direction code must be sorted. The sorting will be performed according to the ascending y-coordinate of the starting position of the direction code.

Only the x or the y-image size, not both, shall exceed the range of 255.

With a bit-depth of 8 bit and resolution of 100 ppcm images of size 2,55 cm × 2,55 cm can be described; covering all available sensors capturing plane impression fingerprint images. For rolled impressions the requirements will be approximately 2,5 cm × 5 cm.

The direction code is sorted by the x-position of its starting point but only the least significant byte of the x coordinate is stored (equal to a mod(256) computation). The card can reconstruct the original sequence of coordinate values by adding 256 to all following entries when a violation of the ascending order occurs. So coordinates with a range of 2,55 cm × infinity can be stored in one byte.

Example

Original sequence:	60	76	277	333	581	797	860	986	1000
Stored sequence:	60	76	21	77	69	29	92	219	231
For each violation of the ascending order add 256 on all following entries:									
+	0	0	256	256	512	768	768	768	768
Reconstructed sequence:	60	76	277	333	581	797	860	986	1000

The most significant byte of x coordinate of the stop position is reconstructed by following the direction code and adding the displacements of each step.

The same construction principle may be applied also for the y coordinate.

8.5 Usage of additional features for the card format

In the card format other features beyond the finger skeletal data may be present. In this case the usage of the biometric data template (tag '7F2E') as described in ISO/IEC 7816-11 and defined in ISO/IEC 7816-6 is mandatory. Table 6 shows the biometric data template with its embedded data objects. If proprietary data are appended, then the biometric data in standardized format (DOs with tag '90' – '93') shall be encapsulated in the DO with tag 'A1'.

Table 6 — Biometric data template

Tag	Length	Value			Presence
'7F2E'	Variable	Biometric data template			
		Tag	**Length**	**Value**	
		'90'	variable	Finger skeletal data according to 8.1 or 8.2, dependent on the indicated format owner/format type	mandatory
		'91'	variable	Ridge cont data according to 7.5.2	optional
		'92'	variable	Core point data according to 7.5.3	optional
		'93'	variable	Delta point data according to 7.5.3	optional
		'94'	variable	Cell quality data according to 7.5.4	optional
		'95'	variable	Sweat pore position data according 7.5.5	optional
		'96'	variable	Skeleton structural data to 7.5.6	optional
		'81'/'A1'	variable	Biometric data with standardized format, see note	optional
		'82'/'A2'	variable	Biometric data with proprietary format	optional
NOTE If the DO with tag '81' is used, then the data according to 8.1 or 8.2 follow without encapsulation.					

8.6 Comparison parameters and card capabilities

Biometric comparison algorithm parameters are used to indicate implementation specific values to be observed by the outside world when computing and structuring the biometric verification data. They can be encoded as DOs embedded in a biometric comparison parameter template as defined in ISO/IEC 19785-1 (see Annex related to smartcards, Table 1).

The comparison parameters and card capabilities for the pattern skeletal format are the maximal data size and the feature handling indicator encoded in the DO 'Biometric algorithm parameters' (tag 'B1' within the BIT, see ISO/IEC 7816-11) (see Table 7).

Table 7 — DO 'Biometric algorithm parameters'

Tag	Length	Value		
'B1'	variable	Biometric algorithm parameters template		
		Tag	**Length**	**Value**
		'81'	2	Maximal data size
		'83'	1	Feature handling indicator, see Table 8

8.6.1 Maximal data size

The maximal data size for the skeleton finger description accepted by a specific card is limited e.g. due to buffer restrictions and computing capabilities.

The maximal data size accepted is therefore an implementation dependent value and shall be indicated using the DO 'Maximal data size' (tag '81', value field 2 bytes). The nesting of this DO in the DO 'Biometric algorithm parameters' is shown in Table 7.

If the length of the finger pattern skeletal data record exceeds the maximum number processible by a card, truncation is necessary. The truncation is a 2 step process. At first, finger skeleton lines of poor quality are eliminated. If still the data length is too large, then truncation shall be made by peeling off skeleton segments from the convex hull of the described area.

For the indication of the maximal data size expected by the card the DO Maximal data size as shown in Table 7 shall be used.

If this DO is not present in the BIT, the maximal data size is unlimited.

8.6.2 Indication of card capabilities

If a card with on-card comparison supports one or more of the additional features, then the capabilities shall be indicated using the DO 'Feature handling indicator' (tag '83', value field 1 byte). The nesting of this DO ' in the DO 'Biometric algorithm parameters' is shown in Table 7, the coding is denoted in Table 8.

Table 8 — Coding of the feature handling indicator

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
							1	Ridge count supported
						1		Core points supported
					1			Delta points supported
				1				Cell quality supported
			1					Sweat pore positions supported
		1						Skeleton structural data supported
x	x							RFU (Default: 0)

8.7 Pattern card format summary

Table 9 is a reference for the fields present in the finger pattern skeletal data card format. Optional extended data formats for ridge counts, core and delta data, zonal quality information and sweat pore position data are not represented here.

Table 9 — Pattern card format summary

Field	Size	Values	Notes
Tag	Variable		Encoded in ASN.1 according Table 6
Length	Variable		Encoded in ASN.1
Skeleton image size in x	2 bytes		in pixels
Skeleton image size in y	2 bytes		in pixels
Length of finger pattern skeletal data	2 bytes		in bytes
Finger pattern skeletal data	In prev. field		
Length of skeleton line neighbourhood index data	2 bytes		In bytes
Skeleton line neighbourhood index data	In prev. field		

9 CBEFF format owner and format types

Format owner and format type are encoded according to CBEFF. The format owner is ISO/IEC JTC 1/SC 37. The IBIA registered format owner id is '????'[1].

The format type denotes one of the finger pattern skeletal formats according to this part of ISO/IEC 19794, see Table 10.

Table 10 — Format types

Format type	Meaning
´0801´	Finger pattern skeletal data record format
´0802´	Finger pattern skeletal data card format – normal size
´0803´	Finger pattern skeletal data card format – compact size

[1] ISO/IEC JTC 1/SC 37 will register with IBIA at completion of FDIS ballot for this part of ISO/IEC 19794.

Annex A
(informative)

Examples for finger pattern skeletal data

The following coding examples have field values according to the recommendations found in Table 5 (which equal those specified in the compact card format as described in clause 8.2): S_s = 16, S_p = 3,75, angular resolution for minutiae and directional elements 5,625°.

The direction dependant step sizes are calculated according to equation (2) in Clause 6.2.4. For low resolution mode:

r_0 = 1,60, r_1 = 1,46, r_2 = 1,31, r_3 = 1,14, r_4 = 0,97, r_5 = 0,78, r_6 = 0,59, r_7 = 0,39 mm.

For high resolution mode:

r_0 = 0,80, r_1= 0,73, r_2 = 0,65, r_3 = 0,57, r_4 = 0,48, r_5 = 0,39, r_6 = 0,30, r_7 = 0,20 mm.

Each of the following examples are represented by a table with a figure in the 1st column. The minutia ID in the 2nd column is displayed as well. Concatenating the byte values in the 6th column of the table gives the finger pattern skeletal data for the ridge structure displayed in the figure.

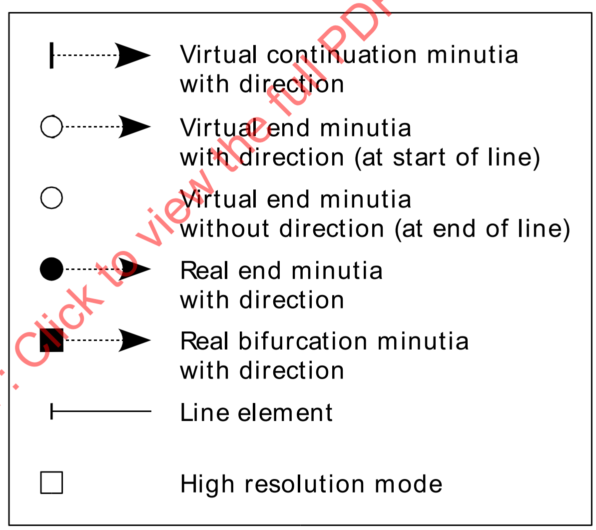
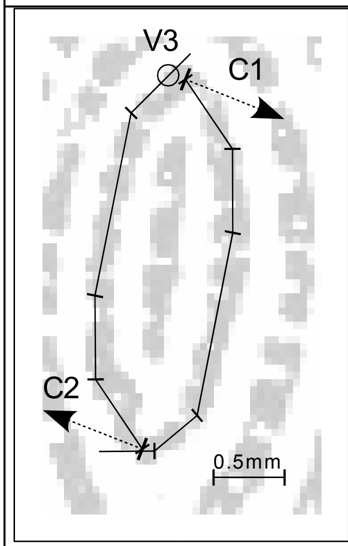


Figure A.1 — Legend for the figures below.

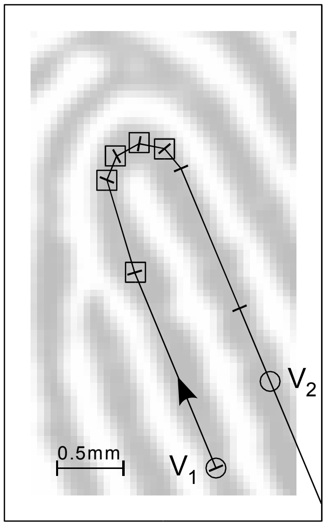
A.1 Virtual continuation

Table A.1 — Virtual continuation minutiae type example. The closed loop without any real minutiae starts at the virtual continuation C_1 and ends with the virtual ending V_3. To describe the high curvature the virtual continuation C_2 is inserted. (Legend for figure see Figure A.1)

Figure	Minutia ID	Type	Value	Bit-depth	Byte value	Absolute direction	Step length [mm]
	C_1	Virtual continuation type	3	2	0xfc		
		Direction	60	6		337,500°	
		x-coordinate	10	8	0x0a		
		y-coordinate	3	8	0x03		
		Number of line elements	5	8	0x05		
		Line element	-6	4	0xaa	303,750°	0,59
		Line element	-6	4		270,000°	0,59
		Line element	-2	4	0xe9	258,750°	1,31
		Line element	-7	4		219,375°	0,39
		Line element	-7	4	0x9c	180,000°	0,39
	C_2	Virtual continuation type	3	2			
		Byte alignment	0	2			
		Virtual continuation type	3	2	0xdc		
		Direction	28	6		157,500°	
		x-coordinate	7	8	0x07		
		y-coordinate	29	8	0x1d		
		Number of line elements	4	8	0x04		
		Line element	-6	4	0xaa	123,750°	0,59
		Line element	-6	4		90,000°	0,59
		Line element	-2	4	0xea	78,750°	1,31
		Line element	-6	4		45,000°	0,59
	V_3	Virtual end type	0	2	0x20		
		Relative pos. on line	2	2			
		Byte alignment	0	4			

A.2 High resolution mode

Table A.2 — High resolution mode example. High resolution mode is used to describe a high curvature line. When beginning the curve description, high-resolution mode is begun with the switch to high resolution (value –8), and returning to full step size after the switch to low resolution (next value –8)(see Clause 6.2.1). (Legend for figures see Figure A.1).

Figure	Minutia ID	Type	Value	Bit-depth	Byte value	Absolute direction	Step length [mm]
	V_1	Virtual end type	0	2	0x14		
		Direction	20	6		112,500°	
		x-coordinate	14	8	0x0e		
		y-coordinate	33	8	0x21		
		Number of line elements	10	8	0x0a		
		Line element	+0	4	0x08	112,500°	1,60
		Switch to high resolution	-8	4			
		Line element	-1	4	0xf9	106,875°	0,73
		Line element	-7	4		67,500°	0,20
		Line element	-7	4	0x99	28,125°	0,20
		Line element	-7	4		348,750°	0,20
		Line element	-7	4	0x98	309,375°	0,20
		Switch to low resolution	-8	4		°	
		Line element	-3	4	0xda	292,500°	1,14
		Line element	-6	4		292,500°	1,60
	V_2	Virtual end type	0	2	0x10		
		Relative pos. on line	1	2			
		Byte alignment	0	4			