Fingerprint identification for embedded platforms

FingerCell SDK
FingerCell technology is designed for embedded biometric systems developers and features compact, sensor-independent and cross-platform fingerprint recognition algorithm. It offers decent performance on various embedded devices based on low-power microcontrollers or processors.

FingerCell is available for integrators as Software Development Kits (SDK) with FingerCell library or source code for developing a fast and reliable system on embedded or mobile platform.

- Fast performance even on low speed processors.
- Verification (1-to-1 matching) and identification (1-to-many matching) are provided.
- Compact fingerprint template and unlimited database size.
- ISO biometric standards support.
- Cross platform algorithm with compact portable source code.
- FingerCell Demo Unit with pre-installed algorithm is optionally available.
- VeriFinger SDK for desktop and mobile platforms is optionally available.
- Reasonable prices, flexible licensing and free customer support.
FingerCell Algorithm Features and Capabilities

FingerCell is designed to provide decent reliability and identification speed for various embedded devices and platforms. The FingerCell algorithm includes these proprietary solutions:

- **Fast performance.** Fingerprint template extraction from an image and verification against another template can be performed in less than less than 0.7 seconds on a 168 MHz ARM Cortex-M4 family processor, which is acceptable for embedded systems.

- **Identification ability.** FingerCell is suitable not only for fingerprint verification (1-to-1 matching), but also for identification (1-to-many matching). The algorithm matches about 250 fingerprints per second in 1-to-many mode on a 168 MHz ARM Cortex-M4 family processor.

- **Adaptive image filtration.** This algorithm eliminates noises, ridge ruptures and stuck ridges for reliable minutiae extraction even from poor quality fingerprints.

- **Compact fingerprint template.** FingerCell template size depends on the number of stored minutiae – for example, a template with 16 minutiae needs only 152 bytes of memory, whereas a template with 64 minutiae needs 448 bytes. Combined with configurable maximal number of minutiae in a template and unlimited database size, the target system size and performance can be optimized according to customers requirements.

- **ISO/IEC standards support.** FingerCell SDK can generate and match fingerprint templates in the ISO/IEC 19794 family formats

- **Tolerance to fingerprint translation and rotation.** Such tolerance is achieved by FingerCell proprietary fingerprint matching algorithm. The algorithm is able to identify fingerprints even if they are rotated and translated.

- **Compact portable software.** FingerCell is designed for easy implementation into very various and specific applications. The algorithm’s source code is sensor independent; therefore it can be ported to various platforms and hardware. Compiled code and internal data arrays require only 128 kB of memory and therefore can be implemented in low memory microchips, thus reducing hardware costs.

- **FingerCell Demo Unit.** Neurotechnology offers pre-installed FingerCell algorithm on testing hardware for the technology evaluation. The Demo Unit is available on request.
System Architecture

Different embedded biometric projects may have specific requirements for system architecture. The components of FingerCell SDK provide interoperability with other Neurotechnology biometric SDKs or third party products and are designed for using in different scenarios:

- **Template extraction and matching on embedded device.** This scenario offers privacy and security, as biometric templates do not leave the device. All functionality can be implemented using only FingerCell SDK and its components, without the need to use any other products. Note, that an embedded device should provide enough computational resources to perform all operations in reasonable time.

- **Template extraction on embedded device, template matching on smart card.** In this scenario, privacy and security is achieved by smart card usage for identity verification, as biometric information is only transferred from embedded device to smart card and is not exposed. Smart card matching technology is not included in the FingerCell SDK. These technologies can be used:
  - *MegaMatcher On Card SDK* is our multi-biometric matching-on-card technology, which is compatible with fingerprint templates generated by FingerCell SDK.
  - Other vendors’ matching on card technologies, which accept biometric templates in ISO/IEC 19794-2 format.

- **Template extraction on embedded device, template matching on server or cloud.** In this scenario, an embedded device, which runs FingerCell algorithm, performs fingerprint template extraction and sends the fingerprint template to a server or cloud for matching. These server-side template matching technologies may be considered, if a system includes large biometric database or should feature high performance:
  - *VeriFinger SDK* and *MegaMatcher SDK* are our biometric identification technologies, which are compatible with fingerprint templates generated by FingerCell SDK and include ready-to-use components for server-side template matching.
  - Other vendors’ server-side fingerprint matching technologies, which accept biometric templates in ISO/IEC 19794-2 or ANSI/INCITS 378 formats.

- **Template extraction on PC or mobile device, template matching on embedded device.** In this scenario, an embedded device, which runs FingerCell algorithm, accepts fingerprint templates for further matching. Fingerprint templates can be generated using these technologies:
  - *VeriFinger SDK* and *MegaMatcher SDK* are our biometric identification technologies, which include components for fingerprint template extraction on Microsoft Windows, macOS, iOS, Android, Linux x86/x86_64 and ARM Linux platforms. The components can be configured to generate fingerprint templates which are compatible with FingerCell SDK.
  - Other vendors’ server-side fingerprint matching technologies, which generate biometric templates in ISO/IEC 19794-2 or ANSI/INCITS 378 formats.
Contents of FingerCell SDK

FingerCell 3.2 SDK is based on FingerCell 3.2 technology that is specially designed for integrating biometric fingerprint recognition into hardware with low-power, low-memory microcontrollers. The fingerprint templates created with FingerCell SDK are compatible with VeriFinger SDK, MegaMatcher SDK and MegaMatcher On Card SDK biometric technologies. Also, FingerCell SDK is compatible with third-party biometric systems, as it accepts and generates fingerprint templates in ISO/IEC 19794-2 and ANSI/INCITS 378 formats.

The following types of FingerCell 3.2 SDK are available:

- **FingerCell 3.2 ARM Linux SDK** – provides the FingerCell components as a shared library, which is compiled for ARM Linux platform. The components are intended to be run on hardware, which meets the system requirements. The SDK also includes documentation with programming samples and tutorials. See also the licensing model.

- **FingerCell 3.2 Library SDK** – provides the FingerCell components as a static library, which is compiled for required platform. The SDK also includes documentation with programming samples and tutorials. See also the licensing model.

- **FingerCell 3.2 Source Code SDK** – provides the FingerCell components as source code, which is intended for porting into required platform. The SDK also includes full documentation for the source code. See also the licensing model.

The FingerCell SDK components provide this functionality:

- **Fingerprint template extraction.** The component creates fingerprint templates from fingerprint images which are provided to the component by integrators. Fingerprint templates can be stored in the following formats:
  - Neurotechnology proprietary fingerprint template format;
  - ISO/IEC 19794-2:2005 with Cor. 1:2009 (Biometric Data Interchange Formats – Finger Minutiae Data (General Record and On-Card Formats)).

- **Fingerprint template stitching.** The component combines multiple fingerprint templates into a single template, which can significantly improve recognition accuracy. The template stitching algorithm is specially designed for use with small area sensors.

- **Fingerprint template matching.** Template matching can be performed in 1-to-1 (verification) and 1-to-many (identification) modes. The component accepts fingerprint templates in the following formats:
  - Neurotechnology proprietary fingerprint template format;
  - ISO/IEC 19794-2:2005 with Cor. 1:2009 (Biometric Data Interchange Formats – Finger Minutiae Data (General Record and On-Card Formats)).
System requirements

There are specific system requirements for evaluating FingerCell technology, developing a FingerCell-based solution and deploying it on embedded hardware.

General deployment platform requirements

There is a list of general requirements for deploying a FingerCell-based software on embedded hardware with low-power microcontrollers. If you are going to use more powerful hardware, like Raspberry Pi, see more specific requirements below.

The requirements are provided for performing operations with 180 x 256 pixels fingerprint images at 385 ppi resolution, or, correspondingly, 234 x 332 pixels at 500 ppi.

- A device with ARM-based microcontroller:
  - ARM Cortex-M4 based microcontroller, running at 168 MHz or better recommended for performing template extraction and matching in the specified time.
  - Floating Point Unit (FPU) is not required for the FingerCell algorithm.
  - Slower microcontrollers may be used if a system uses smaller fingerprint images or has lower performance requirements.

- Memory requirements depend on a specific operation performed with fingerprint templates. Note that RAM is mostly used only during a specific operation (extraction, matching, stitching) and is freed afterwards, so it can be reused for another operation. The program data (code) is intended to be stored and executed in flash memory:
  - Template extraction from an image requires 128 kB of RAM and 100 kB of Flash storage for the 385 ppi image specified above.
  - Template matching requires 16 kB of RAM and 70 kB of Flash storage.
  - Template stitching requires 50 kB of RAM and 80 kB of Flash storage. This RAM amount is required for performing the operation with 9 templates.
  - Additional flash storage is required for systems that store multiple fingerprint templates.
  - Additional RAM is required in these cases:
    - The original raw fingerprint image needs to be preserved in the RAM during the template extraction operation. In this case the additional amount of RAM equals to the image bitmap size (i.e. additional 45 kB for a 180 x 256 pixels image).
    - The raw fingerprint image is larger than specified above. In this case the minimal amount of RAM for template extraction will be equal to 2 times the image bitmap size plus 10 kB for internal data structures. This amount does not include the space for preserving the original image in the RAM – see the comment above.
    - The system performs 1-to-many identification, as all biometric templates need to be stored in RAM for matching. See technical specifications for more information.
    - The RAM should be not fragmented – at least it should have continuously addressable space to fit single copy of raw image.

continued on the next page.
FingerCell technology can be deployed on different platforms, which can be with or without operating system. However, FingerCell libraries require some functions from the standard C library: malloc, calloc, realloc, free, memcpy, memcmp, memset, memmove, qsort, pow. These functions should be provided by the integrators.

- **Fingerprint readers and fingerprint images.** The FingerCell Extractor component directly accepts fingerprint images as raw grayscale pixels for further biometric template extraction, thus almost any fingerprint sensor can be used.
  - Integrators should implement by themselves the passing of fingerprint images to a device which runs the FingerCell algorithm.
  - The fingerprint images should meet the technical specifications for acceptable fingerprint recognition performance on a target device.

**ARM Linux specific deployment platform requirements**

Please contact us and report the specifications of a target device to find out if it will be suitable for running FingerCell-based applications. See the general deployment requirements above on this page for a different hardware platform.

The FingerCell ARM Linux SDK is intended for deploying the FingerCell technology on Raspberry Pi single-board computers or similar devices. The detailed requirements are:

- A device with ARM-based processor, running Linux 3.2 kernel or newer. Raspbian Linux distribution is recommended.
- ARM-based 900 MHz processor or better is recommended. Floating Point Unit (FPU) is not required for the FingerCell algorithm.
- At least 2 MB of free RAM should be available for the FingerCell algorithm.
  - Additional RAM is required if the system performs 1-to-many identification, as all biometric templates need to be stored in RAM for matching. See technical specifications for more information.
- **Fingerprint reader.** FingerCell SDK includes support modules for several fingerprint scanners under ARM Linux platform. Also, fingerprint images in BMP, JPG or PNG formats can be processed thus almost any third-party fingerprint capturing hardware can be used with the FingerCell technology if it generates images in the mentioned formats.
  - glibc 2.13 or newer.
  - libstdc++-v3 4.7.2 or newer.

*continued on the next page.*
Technology evaluation and development platform requirements

At the moment FingerCell technology can be evaluated on Microsoft Windows and Linux platforms. There are some specific requirements for developing FingerCell-based applications, as well as running FingerCell technology demo application on Microsoft Windows:

- PC or laptop with x86 (32-bit) or x86-64 (64-bit) compatible processors.
  - 2 GHz or better processor is recommended.
  - SSE2 support is required. Processors that do not support SSE2 cannot run the FingerCell algorithm. Please check if a particular processor model supports SSE2 instruction set.
  - If a fingerprint scanner is required, note that some scanners are supported only on 32-bit OS or only from 32-bit applications.
- At least 128 MB of free RAM should be available for the application. Additional RAM is required for applications that perform 1-to-many identification, as all biometric templates need to be stored in RAM for matching. For example, 10,000 templates (each with 1 fingerprint inside) require from 10 MB of additional RAM depending on configured template size.
- Fingerprint reader (optional). The trial version of FingerCell SDK includes support modules for more than 100 fingerprint scanners and sensors under Microsoft Windows platform. Also, fingerprint images in BMP, JPG or PNG formats can be provided to the FingerCell algorithm for evaluation.
- Network/LAN connection (TCP/IP) for client/server applications. If communication must be secured, a dedicated network (not accessible outside the system) or a secured network (such as VPN; VPN must be configured using operating system or third party tools) is recommended.
- Microsoft Windows specific:
  - Microsoft Windows 7 / 8 / 10, 32-bit or 64-bit.
  - Microsoft .NET framework 4.5 or newer (for .NET components usage).
- Linux x86/x86_64 specific:
  - Linux 2.6 or newer kernel (32-bit or 64-bit) is required. Linux 3.0 kernel or newer is recommended.
  - glibc 2.13 library or newer
  - GCC-4.4.x or newer
  - GNU Make 3.81 or newer
**Technical Specifications**

385 ppi is the minimal recommended fingerprint image resolution for FingerCell template extraction algorithm.

If the system needs to perform person’s identification (1-to-many matching), all fingerprint templates should be loaded into RAM, thus the maximum fingerprint templates database size is limited by the amount of available RAM. See system requirements on the previous page for more information about the required amounts of RAM and flash storage.

The performance specifications below are provided for embedded hardware based on ARM Cortex-M4F microcontroller, running at 168 MHz clock rate.

<table>
<thead>
<tr>
<th>FingerCell 3.2 algorithm technical specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template extraction time (milliseconds) (^{(1)})</td>
</tr>
<tr>
<td>Template stitching time (milliseconds) (^{(2)}) (^{(3)})</td>
</tr>
<tr>
<td>Template verification time (milliseconds) (^{(3)})</td>
</tr>
<tr>
<td>Template identification speed (templates per second) (^{(3)})</td>
</tr>
<tr>
<td>Template size with 16 minutiae (bytes) (^{(4)})</td>
</tr>
<tr>
<td>Template size with 64 minutiae (bytes) (^{(4)})</td>
</tr>
</tbody>
</table>

Notes:
1. For performing the operation with 180 x 256 pixels fingerprint images at 385 ppi resolution, or, correspondingly, 234 x 332 pixels at 500 ppi.
2. For performing the operation with 9 fingerprint templates.
3. For templates containing up to 64 minutiae.
4. The template size depends on the actual number of minutiae stored in it. The provided values are reference sizes for the corresponding numbers of minutiae.
FingerCell Demo Unit

Neurotechnology offers FingerCell Demo Unit – testing hardware with pre-installed FingerCell algorithm for the technology evaluation. Please contact us for more information about obtaining the Demo Unit.

FingerCell Demo Unit is based on an STM Discovery evaluation board with these components:

- STM32F407 microcontroller running at 168 MHz and having 192 kB of RAM;
- Next Biometrics fingerprint sensor connected via SPI, which produces 180 x 256 pixels grayscale image at 385 ppi resolution;
- 2x16 LCD screen;
- 4 push buttons.

The FingerCell algorithm in the Demo Unit performs single fingerprint template extraction in 700 milliseconds and matches 250 fingerprints per second.

The Demo Unit can be used in two modes:

- Connected to a host PC – all operations, including template import and export, are performed via USB communication.
  - PC-side API and a programming sample, which shows how to use all commands, are provided.
  - The host PC needs to run Linux OS.
- Stand-alone – these actions are performed by the unit after pressing the corresponding buttons:
  - ENROLL – performs single fingerprint capture and enrollment. The LCD screen indicates that enrollment is in progress and asks to put a finger on scanner. Once the finger is captured, the template will be extracted, duplicates will be checked in the database and the finger will be enrolled if no duplicates are found. An auto-generated ID for the enrolled fingerprint will be displayed.
  - ENROLL-STITCH – performs multiple capture of the same fingerprint, template stitching and enrollment. A user should scan his/her finger 8 times. The LCD screen will show the enrollment progress, and a template will be generated for each scanned fingerprint. After 8 scans, all templates will be merged into single generalized template, which will be enrolled into database and the auto-generated ID will be displayed.
  - IDENTIFY – performs single fingerprint capture and identification against the templates stored in the database. A user should scan a fingerprint, which will be processed and identified against the previously enrolled fingerprints. The LCD screen will show the identification process progress. If match a is found, the matching fingerprint ID and matching score will be displayed. The higher matching score is, the more similar the fingerprints are.
  - CANCEL/CLEAR – single click cancels the selected operation, and double click deletes all entries from the database.
FingerCell SDK Trial and Related Products

Neurotechnology offers the 30 day trial version of FingerCell SDK for developers and integrators. The trial version is provided only for evaluating the FingerCell technology capabilities on Microsoft Windows platform, whereas the FingerCell algorithm is designed for running on embedded devices and platforms based on low-power microcontrollers or processors, with or without operating system.

FingerCell SDK 30-day trial can be downloaded at www.neurotechnology.com/download.html

These products are related to FingerCell EDK (see the corresponding product brochure for more information):

- **MegaMatcher On Card SDK** – a product for fingerprint, iris and face matching on smart cards. Biometric System On Card can be implemented, as fingerprint templates, which were generated with FingerCell extractor, can be sent for matching to a smart card, which runs MegaMatcher On Card algorithm.

- **VeriFinger SDK** – for developing stand-alone or web-based fingerprint identification solutions on Microsoft Windows, macOS, iOS, Android and Linux platforms. Fingerprint templates, which were generated with FingerCell extractor, can be used in a VeriFinger-based solution.
Licensing FingerCell EDK

Product Development
An integrator should obtain FingerCell 3.2 ARM Linux SDK to develop an end-user product based on the FingerCell technology. The SDK needs to be purchased just once and may be used for all projects and by all the developers within the integrator’s company.

FingerCell 3.2 ARM Linux SDK includes 500 single computer licenses for the FingerCell component for ARM Linux devices.

Integrators can obtain additional component licenses if more component licenses are required for the development process.

Product Deployment
To deploy their developed products, an integrator needs to obtain licenses of components for every device, where the component will be installed together with integrator’s product. Integrators can purchase additional FingerCell component licenses if required at anytime.

License activation options
The components are copy-protected. The following license activation options are available:

- **Internet activation.** A special license file is stored on a computer or a mobile or embedded device; the license file allows to run particular FingerCell components on that device after checking the license over the Internet. Internet connection should be available periodically for a short amount of time. A single computer license can be transferred to another computer or device by moving the license file there and waiting until the previous activation expires.

- **Volume License Manager.** Licenses may be stored in a volume license manager dongle. License activation using volume license manager may be performed without connection to the Internet and is suitable for virtual environments. Volume license manager is used on site by integrators or end users to manage licenses for FingerCell components in the following ways:
  1. **Activating single computer licenses** – An installation license for a FingerCell component will be activated for use on a particular computer. The number of available licenses in the license manager will be decreased by the number of activated licenses.
  2. **Managing single computer licenses via a LAN or the Internet** – The license manager allows the management of installation licenses for FingerCell components across multiple computers or mobile/embedded devices in a LAN or over the Internet. The number of managed licenses is limited by the number of licenses in the license manager. No license activation is required and the license quantity is not decreased. Once issued, the license is assigned to a specific computer or device on the network.
  3. **Using license manager as a dongle** – A volume license manager containing at least one license for a FingerCell component may be used as a dongle, allowing the FingerCell component to run on the particular computer where the dongle is attached.

continued on the next page.
Licenses Validity
All SDK and component licenses are perpetual and do not have expiration. There are no annual fee or any other fees except license purchasing fee. It is possible to move licenses from one computer or device to another. Neurotechnology provides a way to renew the license if the computer undergoes changes due to technical maintenance.

Licensing Agreement

Note that you unambiguously accept this agreement by placing an order using Neurotechnology online ordering service or by email or other means of communications. Please read the agreement before making an order.

Library, Library Enterprise and Source code licenses
FingerCell 3.2 SDK is offered in the following options:

- **Library license.** The Library license is a royalty based license which allows usage of FingerCell static library for the end user products as long as required amount of installation licenses is obtained. The installation licenses are per embedded device, however, no activation on particular device is needed – FingerCell usage and deployment are controlled by licensing agreement. The installation licenses are obtained in advance and their usage is reported.

- **Library Enterprise license.** The Library Enterprise license allows unlimited usage of FingerCell static library for the end user products for just one time EUR 150,000 payment.

- **Source code license.** The Source code license covers source code of FingerCell algorithm and allows unlimited usage of FingerCell for the end user products for just one time EUR 250,000 payment.

FingerCell 3.2 SDK prices can be found on the next page.

Integrators should sign the FingerCell 3.2 SDK Software Licensing Agreement before purchasing FingerCell 3.2 SDK library or source code packages.
**Prices**

- These prices are effective January 7, 2019. The prices may change in the future, so please download and review the latest version of the brochure before making an order.
- Prices do not include local import duties or taxes.

**FingerCell ARM Linux SDK**

<table>
<thead>
<tr>
<th>FingerCell 3.2 ARM Linux SDK package (includes 500 FingerCell 3.2 installation licenses, larger quantities can be also ordered – see the prices below)</th>
<th>€ 990.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional FingerCell 3.2 installation licenses</td>
<td></td>
</tr>
<tr>
<td>Quantity</td>
<td>Price per device</td>
</tr>
<tr>
<td>100 - 199</td>
<td>€ 1.87</td>
</tr>
<tr>
<td>200 - 499</td>
<td>€ 1.68</td>
</tr>
<tr>
<td>500 - 999</td>
<td>€ 1.46</td>
</tr>
<tr>
<td>1,000 - 1,999</td>
<td>€ 1.31</td>
</tr>
<tr>
<td>2,000 - 3,999</td>
<td>€ 1.17</td>
</tr>
<tr>
<td>4,000 - 8,999</td>
<td>€ 1.05</td>
</tr>
<tr>
<td>8,000 - 15,999</td>
<td>€ 0.95</td>
</tr>
<tr>
<td>16,000 - 31,999</td>
<td>€ 0.85</td>
</tr>
<tr>
<td>32,000 - 63,999</td>
<td>€ 0.76</td>
</tr>
<tr>
<td>64,000 - 127,999</td>
<td>€ 0.68</td>
</tr>
</tbody>
</table>

**Notes:**
1. The minimal number of additional licenses which can be purchased is 100 licenses.
2. Quantity discounts are applied for the current order and do not accumulate over time.

*continued on the next page.*
### FingerCell Library license

**FingerCell Library license price consists of the following parts:**

<table>
<thead>
<tr>
<th><strong>FingerCell 3.2 Library SDK package</strong></th>
<th>€ 50,000.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>(includes 70,000 FingerCell 3.2 installation licenses, larger quantities can be also ordered – see the prices below)</td>
<td></td>
</tr>
</tbody>
</table>

**Library preparation for the particular platform and other custom works**
- Depends on the required time, contact us for more information

**Additional FingerCell 3.1 installation licenses**

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Price per device</th>
</tr>
</thead>
<tbody>
<tr>
<td>16,000 - 31,999</td>
<td>€ 0.85</td>
</tr>
<tr>
<td>32,000 - 63,999</td>
<td>€ 0.76</td>
</tr>
<tr>
<td>64,000 - 127,999</td>
<td>€ 0.68</td>
</tr>
<tr>
<td>128,000 - 255,999</td>
<td>€ 0.61</td>
</tr>
<tr>
<td>256,000 - 511,999</td>
<td>€ 0.55</td>
</tr>
</tbody>
</table>

**Notes:**
1. The minimal number of licenses which can be purchased is 16,000 licenses.
2. Quantity discounts are applied for the current order and do not accumulate over time.

### FingerCell Library Enterprise license

**FingerCell Library Enterprise license**

<table>
<thead>
<tr>
<th><strong>FingerCell 3.2 Library Enterprise license</strong></th>
<th>€ 150,000.00</th>
</tr>
</thead>
</table>

### FingerCell Source Code license

**FingerCell Source code license**

<table>
<thead>
<tr>
<th><strong>FingerCell 3.2 Source code license</strong></th>
<th>€ 250,000.00</th>
</tr>
</thead>
</table>