



NEUROtechnology



Face identification
for stand-alone
or Web
applications

VeriLook SDK



VeriLook SDK

Face identification for stand-alone or Web applications

Document updated on **June 20, 2016**

CONTENTS

VeriLook algorithm features and capabilities	3
Contents of VeriLook 9.0 Standard SDK and Extended SDK	4
Biometric components description.	5
Basic recommendations for facial recognition	10
Supported cameras	12
System requirements.	13
Technical specifications	19
Reliability tests.	20
VeriLook Demo, Trial SDK and related products	22
Licensing VeriLook SDK	23
Prices for VeriLook products	26

VeriLook facial identification technology is designed for biometric systems developers and integrators. The technology assures system performance and reliability with live face detection, simultaneous multiple face recognition and fast face matching in 1-to-1 and 1-to-many modes.

VeriLook is available as a software development kit that allows development of stand-alone and Web-based solutions on Microsoft Windows, Linux, Mac OS X, iOS and Android platforms.

- More than a million algorithm deployments worldwide.
- Live face detection prevents cheating with a photo in front of a camera.
- Simultaneous multiple face processing in live video and still images.
- Gender recognition and facial feature points extraction for each person in an image.
- Webcams or other low cost cameras are suitable for obtaining face images.
- Available as multiplatform SDK that supports multiple programming languages.
- Surveillance SDK is available for integrating face identification into surveillance systems.
- Reasonable prices, flexible licensing and free customer support.



VeriLook algorithm features and capabilities

Performance numbers are provided for a PC with Intel Core 2 Q9400 processor (2.67 GHz).

The VeriLook algorithm implements advanced face localization, enrollment and matching using robust digital image processing algorithms:

- **Simultaneous multiple face processing.** VeriLook 9.0 performs fast and accurate detection of multiple faces in **live video** streams and still images. All faces on the current frame are detected in **0.01 - 0.86 seconds** depending on selected values for face roll and yaw tolerances, and face detection accuracy. After detection, a set of features is **extracted** from each face into a template in 0.6 seconds. See technical specifications for more details.
- **Gender classification.** Optionally, gender can be determined for each person on the image with predefined degree of accuracy during the template extraction.
- **Live face detection.** A conventional face identification system can be tricked by placing a photo in front of the camera. VeriLook is able to prevent this kind of security breach by determining whether a face in a video stream is “live” or a photograph. See recommendations for live face detection in the chapters below for more details.
- **Emotions recognition.** VeriLook can be configured to recognize emotion type in a human face. Six basic emotions are analyzed: anger, disgust, fear, happiness, sadness and surprise. A confidence value for each of the basic emotions is returned for the face. Larger value for an emotion means that it seems to be more expressed in the face
- **Facial feature points.** The points can be optionally extracted as a set of their coordinates during the face template extraction. Each facial feature point has a fixed sequence number (i.e. number 31 always corresponds to nose tip).
- **Facial attributes.** VeriLook can be configured to detect certain attributes during the face extraction – **smile, open-mouth, closed-eyes, glasses, dark-glasses, beard and mustache.**
- **Age estimation.** VeriLook can optionally estimate person’s age by analyzing the detected face in the image.
- **Face image quality determination.** A quality threshold can be used during face enrollment to ensure that only the best quality face template will be stored into database.
- **Tolerance to face position.** VeriLook allows for 360 degrees of head roll. Head pitch can be up to 15 degrees in each direction from the frontal position. Head yaw can be up to 45 degrees in each direction from the frontal position. See technical specifications for more details.
- **Multiple samples of the same face.** Biometric template record can contain multiple face samples belonging to the same person. These samples can be enrolled from different sources and at different times, thus allowing improvement in matching quality. For example, a person might be enrolled with and without beard or mustache, etc.
- **Identification capability.** VeriLook functions can be used in 1-to-1 matching (verification), as well as 1-to-many mode (identification). The VeriLook 9.0 face template matching algorithm can compare up to 40,000 faces per second on a PC. See technical specifications for more details.
- **Small face features template.** A face features template can be only **4 Kilobytes**, thus VeriLook-based applications can handle **large face databases**. Also, 5 Kilobytes and 7 Kilobytes templates can be used to increase matching reliability. See technical specifications for more details.
- **Features generalization mode.** This mode generates the collection of the generalized face features from several images of the same subject. Then, each face image is processed, features are extracted, and the collections of features are analyzed and combined into a single generalized features collection, which is written to the database. This way, the enrolled feature template is more reliable and the face recognition quality increases considerably.



Contents of VeriLook 9.0 Standard SDK and Extended SDK

VeriLook SDK is based on VeriLook face recognition technology and is intended for biometric systems developers and integrators. The SDK allows rapid development of biometric applications using functions from the VeriLook algorithm that ensure fast and reliable face identification. VeriLook can be easily integrated into the customer's security system. The integrator has complete control over SDK data input and output.

VeriLook SDK includes the Device Manager library that allows to perform **simultaneous capture from multiple cameras**. Integrators can write **plug-ins to support their cameras** or other devices using the plug-in framework provided with the Device Manager.

VeriLook is available as the following SDKs:

- **VeriLook 9.0 Standard SDK** is designed for PC-based, embedded or mobile biometric application development. It includes Face Matcher, Face Extractor and Face Verification component licenses, programming samples and tutorials, Device Manager library and software documentation. The SDK allows the development of biometric applications for Microsoft Windows, Linux, Mac OS X, iOS or Android operating systems.
- **VeriLook 9.0 Extended SDK** is designed for biometric **Web-based** and network application development. It includes all features and components of the Standard SDK. Additionally, the SDK contains Face Client component licenses for PCs and mobile / embedded devices, sample client applications, tutorials and a **ready-to-use matching server** component.

The table below compares VeriLook 9.0 Standard SDK and VeriLook 9.0 Extended SDK. See the licensing model for more information on specific license types.

Component licenses that are included with a specific SDK		
	VeriLook 9.0 Standard SDK	VeriLook 9.0 Extended SDK
• Face Verification component	1 single computer license	1 single computer license
• Face Matcher	1 single computer license	1 single computer license
• Embedded Face Matcher for Android	1 single computer license	1 single computer license
• Embedded Face Matcher for iOS	1 single computer license	1 single computer license
• Embedded Face Matcher for ARM Linux	1 single computer license	1 single computer license
• Face Client ⁽¹⁾		3 single computer licenses
• Embedded Face Client for Android		3 single computer licenses
• Embedded Face Client for iOS		3 single computer licenses
• Embedded Face Client for ARM Linux		3 single computer licenses
• Face Extractor	1 single computer license	1 single computer license
• Embedded Face Extractor for Android	1 single computer license	1 single computer license
• Embedded Face Extractor for iOS	1 single computer license	1 single computer license
• Embedded Face Extractor for ARM Linux	1 single computer license	1 single computer license
• Matching Server		+

(1) Face Client component includes Face Extractor, Face Token Image and Face BSS components, which can be also obtained separately.

VeriLook 9.0 SDK includes programming samples and tutorials that show how to use the components of the SDK to perform face template extraction or matching against other templates. The samples and tutorials are available for these programming languages and platforms:

Programming samples and tutorials					
	Windows 32 & 64 bit	Linux 32 & 64 bit	Mac OS X	Android	iOS
• C/C++	+	+	+		
• Objective-C					+
• C#	+				
• Sun Java 2	+	+	+	+	
• Visual Basic .NET	+				



Biometric Components Description

Face Verification component

The Face Verification component is designed for simple integration of facial recognition technology into high-security applications, like mobile banking transactions, which need only biometric identity verification. The component is intended to provide its functionality for reasonable price, especially for large-scale deployments.

The following operations are available via the high-level API:

- Face **enrollment** into the internal database – an image with a face is captured from a camera, the face template is extracted from the image and saved into the database. Custom meta-information (like person's name) can be provided during calling this operation to store it in the database together with the face template.
- Face **verification** against a specific face from the database – an image with a face is captured from a camera, the face template is extracted from the image and matched against the template stored in the specified database record.
- Database record removal.

Integrators can enable or disable face **liveness detection** to prevent fraud attempts with a photo. Also, parameters like template size or matching quality threshold can be modified before calling the enrollment or verification operations.

One Face Verification component license is included with VeriLook 9.0 Standard SDK and VeriLook 9.0 Extended SDK for all supported desktop and mobile platforms. More licenses for this component can be purchased any time by VeriLook 9.0 SDK customers.

Face Matcher

The Face Matcher performs facial template matching in 1-to-1 (verification) and 1-to-many (identification) modes. Also the Face Matcher component includes **fused** matching algorithm that allows to increase template matching reliability by matching templates that contain fingerprint, face, voiceprint and/or iris records (note that matching fingerprint, irises and voiceprints requires to purchase Fingerprint Matcher, Iris Matcher and Voice Matcher components correspondingly - these components are available in *VeriFinger 9.0 SDK*, *VeriEye 9.0 SDK* and *VeriSpeak 9.0 SDK* correspondingly; see these products brochures for more information).

The Face Matcher component matches **40,000 faces per second** and is designed to be used in **desktop** or mobile biometric systems, which run on PCs or laptops with at least Intel **Core 2 Q9400** (2.67 GHz) processor.

One Face Matcher license is included with VeriLook 9.0 Standard SDK and VeriLook 9.0 Extended SDK. More licenses for this component can be purchased any time by VeriLook 9.0 SDK customers.

Embedded Face Matcher

The Embedded Face Matcher has the same functionality, as the Face Matcher. It matches **3,000 faces per second** and is designed to be used in **embedded** or **mobile** biometric systems, which run on ARM Linux, **Android** or **iOS** devices. The Android devices should be based on at least **Snapdragon S4** system-on-chip (**Krait 300** processor with 4 cores running at 1.51 GHz).

One Embedded Face Matcher license for each of Android, iOS and ARM Linux platforms is included with VeriLook 9.0 Standard SDK and VeriLook 9.0 Extended SDK. More licenses for this component can be purchased any time by VeriLook 9.0 SDK customers.



Face Client

The Face Client component is a combination of the Face Extractor, Face Token Image and Face BSS components. It is designed for the systems that need to support all functionality of the mentioned components on the same PC. Using these licenses allows to optimize component license costs as well as reduce license management.

The Face Client extracts a single face template in **0.6 seconds**. The specified performance requires a **PC or laptop** with at least Intel **Core 2 Q9400** (2.67 GHz) processor.

Three licenses for the Face Client component are included with VeriLook 9.0 Extended SDK. More licenses for this component can be purchased any time by VeriLook 9.0 Extended SDK customers.

Embedded Face Client

The Embedded Face Client component has the same functionality as the Face Client and is designed to run on **Android** or **iOS** or **ARM Linux** devices. The Android devices should be based on at least **Snapdragon S4** system-on-chip (**Krait 300** processor with 4 cores running at 1.51 GHz). The component extracts a single face template in **1.34 seconds**.

Three licenses for the Embedded Face Client component for each of Android, iOS and ARM Linux platforms are included with VeriLook 9.0 Extended SDK. More licenses for this component can be purchased any time by VeriLook 9.0 Extended SDK customers

Face Extractor

Face Extractor creates face templates from face images. The Extractor can generalize a face template from several images that include the same face to improve the template's quality.

Device Manager software allows to perform **simultaneous capture from multiple cameras**. Integrators can write **plug-ins to support their cameras** or other devices using the plug-in framework provided with the Device Manager.

The component extracts a single face template in **1.34 seconds**. The specified performance requires a **PC or laptop** with at least Intel **Core 2 Q9400** (2.67 GHz) processor.

One Face Extractor license is included with VeriLook 9.0 Standard SDK and VeriLook 9.0 Extended SDK. More licenses for this component can be purchased any time by VeriLook 9.0 SDK customers.

Embedded Face Extractor

The Embedded Face Extractor has the same functionality as the Face Extractor and is designed to be run on **Android** or **iOS** or **ARM Linux** devices. The Android devices should be based on at least **Snapdragon S4** system-on-chip (**Krait 300** processor with 4 cores running at 1.51 GHz). The component extracts a single face template in **1.34 seconds**.

One Embedded Face Extractor license for each of Android, iOS and ARM Linux platforms is included with VeriLook 9.0 Standard SDK and VeriLook 9.0 Extended SDK. More licenses for this component can be purchased any time by VeriLook 9.0 SDK customers.



Face Token Image

The Face Token Image component is designed to provide token* face images compatible with the Face Image Format as in ISO/IEC 19794 standard. This face image format enables range of applications on variety of devices, including devices that have limited resources required for data storage, and improves recognition accuracy by specifying data format, scene constraints (lighting, pose), photographic properties (positioning, camera focus) and digital image attributes (image resolution, image size).

The Face Token Image component has the following features:

- Face Token Image creation from an image containing human face using eye coordinates which may be either hand marked or detected automatically using Neurotechnology face detection algorithm.
- Face is detected and eye coordinates are acquired using state-of-the-art Neurotechnology face detection and recognition algorithm.
- Geometrical face image normalization according to the proportions and photographic properties, which are specified in ISO/IEC 19794 standard.
- Intelligent image padding algorithm for cutting off parts of Face Token Image as specified in ISO/IEC 19794 standard.
- Evaluation of the created token face image for the following quality criteria suggested in ISO/IEC 19794 standard:
 - Background uniformity – the background in the token face image should be uniform, not cluttered.
 - Sharpness – the token face image should not be blurred.
 - Too light or too dark images – the token face image should not be too dark or too light.
 - Exposure range of an image – the token face image should have a reasonable exposure range to represent as much details of the subject in the image as possible.
- Evaluation of the token face image quality based on suggestions of ISO/IEC 19794 standard (using the quality criteria above)

The Face Token Image component also includes proprietary algorithms for this functionality:

- Person's gender recognition.
- Emotions detection.
- Facial feature points extraction for each person from an image.
- Age estimation for each person from an image.
- Additional face attributes detection: smile, open-mouth, closed-eyes, glasses and dark-glasses.
- Live face detection can be used for determining whether a face in a video stream belongs to a real human or is a photo. See recommendations for live face detection for more information.

The component is designed for desktop or mobile applications that run on **PC or laptop** with at least Intel **Core 2 Q9400** (2.67 GHz) processor. It can be used from **C/C++**, **C#** and **Java** applications on all supported platforms. **.NET wrappers** of Windows libraries are provided for .NET developers.

Licenses for the Face Token Image component can be purchased anytime by VeriLook 9.0 Extended SDK customers.

**Token in this context is used as "symbolic image, good enough image for machine recognition". Token Image as in ISO/IEC19794-5: "A Face Image Type that specifies frontal images with a specific geometric size and eye positioning based on the width and height of the image. This image type is suitable for minimizing the storage requirements for computer face recognition tasks such as verification while still offering vendor independence and human verification (versus human examination which requires more detail) capabilities."*



Face BSS (Biometric Standards Support)

The Face BSS (Biometric Standards Support) component allows to integrate support for facial image format standards and additional image formats with new or existing biometric systems based on VeriLook SDK.

These biometric standards are supported:

- **BioAPI 2.0 (ISO/IEC 19784-1:2006)** (Framework and Biometric Service Provider for Face Identification Engine)
- **CBEFF V1.2 (ANSI INCITS 398-2008)** (Common Biometric Exchange Formats Framework)
- **CBEFF V2.0 (ISO/IEC 19785-1:2006, 19785-3:2007)** (Common Biometric Exchange Formats Framework)
- **ISO/IEC 19794-5:2005** (Face Image Data)
- **ISO/IEC 19794-5:2011** (Face Image Data)
- **ANSI/INCITS 385-2004** (Face Recognition Format for Data Interchange)
- **ANSI/NIST-CSL 1-1993** (Data Format for the Interchange of Fingerprint, Facial, & SMT Information)
- **ANSI/NIST-ITL 1a-1997** (Data Format for the Interchange of Fingerprint, Facial, & SMT Information)
- **ANSI/NIST-ITL 1-2000** (Data Format for the Interchange of Fingerprint, Facial, & SMT Information)
- **ANSI/NIST-ITL 1-2007** (Data Format for the Interchange of Fingerprint, Facial, & Other Biometric Information)
- **ANSI/NIST-ITL 1a-2009** (Data Format for the Interchange of Fingerprint, Facial, & Other Biometric Information)

Face BSS component also allows to integrate **JPEG 2000** with Lossy and Lossless Face Profiles support into applications based on VeriLook SDK.

Licenses for the Face BSS component can be purchased anytime by VeriLook 9.0 Extended SDK customers.



Matching Server

The Matching Server is ready-to-use software intended for building moderate size web-based and other network-based systems like local single- or multi-biometric identification system. The Server software runs on a server PC and allows to perform the biometric template matching on server side using Face Matcher component.

Fused multi-biometric matching can be enabled by running components for fingerprint, face and iris matching on the same machine.

Client communication module that allows sending a task to the Matching Server, querying status of the task, getting the results and removing the task from server, is included with MegaMatcher 9.0 SDK, VeriFinger 9.0 SDK, VeriLook 9.0 SDK, VeriSpeak 9.0 SDK and VeriEye 9.0 SDK. This module hides all low level communications and provides high-level API for the developer.

The components and database support modules with source codes included for Matching Server component are listed in the table below. Custom modules for working with other databases can also be developed by integrator and used with the Matching Server software.

The table below shows what components are available with Matching Server software.

Components	Microsoft Windows 32 & 64 bit	Linux 32 & 64 bit	Mac OS X
• Matching server software	+	+	+
• Server administration tool API	+	+	
Database support modules			
• Microsoft SQL Server	+		
• PostgreSQL	+	+	
• MySQL	+	+	
• Oracle	+	+	
• SQLite	+	+	+
Programming samples			
• C# client	+		
• Visual Basic .NET client	+		
• Sun Java 2 web client	+	+	+
Programming tutorials			
• C/C++	+	+	
• C#	+		
• Visual Basic .NET	+		

The Matching Server component requires a **special license** that allows to run the component on all machines that run the fingerprint, face, iris or palm print matching components obtained by an integrator. The Matching Server software is included with VeriLook 9.0 Extended SDK.

Also the Matching Server component is included with these Neurotechnology SDKs (see their brochures for more info):

- MegaMatcher 9.0 Standard or MegaMatcher 9.0 Extended SDK;
- VeriFinger 9.0 Extended SDK;
- VeriEye 9.0 Extended SDK.
- VeriSpeak 9.0 Extended SDK.



Basic Recommendations for Facial Recognition

Face recognition accuracy of VeriLook heavily depends on the quality of a face image. **Image quality during enrollment is important**, as it influences the quality of the face template.

General recommendations

- **32 pixels is the recommended minimal distance between eyes** for a face on image or video stream to perform face template extraction reliably. **64 pixels or more** recommended for better face recognition results. Note that this distance should be **native**, not achieved by resizing an image.
- **Several images during enrollment** are recommended for better facial template quality which results in improvement of recognition quality and reliability.
- **Additional enrollments** may be needed when **facial hair** style changes, especially when beard or mustache is grown or shaved off.

Face posture

The face recognition engine has certain tolerance to face posture:

- head **roll** (tilt) – ± 180 degrees (configurable).
 - **± 15 degrees default** value is the fastest setting which is usually sufficient for most near-frontal face images.
- head **pitch** (nod) – ± 15 degrees from frontal position.
 - The head pitch tolerance can be increased up to ± 25 degrees if several views of the same face that covered different pitch angles were used during enrollment.
- head **yaw** (bobble) – ± 45 degrees from frontal position (configurable).
 - **± 15 degrees default** value is the fastest setting which is usually sufficient for most near-frontal face images.
 - **30 degrees difference** between a face template in a database and a face image from camera is **acceptable**.
 - Several views of the same face can be enrolled to the database to cover the whole ± 45 degrees yaw range from frontal position.

Continued on the next page



Live Face Detection

A stream of consecutive images (usually a video stream from a camera) is required for face liveness check:

- When the liveness check is enabled, it is performed by the face engine before feature extraction. If the face in the stream **fails** to qualify as “live”, the features are **not extracted**.
- Only **one face should be visible** in these frames.
- Users can enable these liveness check modes:
 - **Active** – the engine requests the user to perform certain actions like blinking or moving one’s head.
 - 5 frames per second or better frame rate required.
 - This mode can work with both colored and grayscale images.
 - This mode requires the user to perform all requested actions to pass the liveness check.
 - **Passive** – the engine analyzes certain facial features while the user stays still in front of the camera for a short period of time.
 - Colored images are required for this mode.
 - 10 frames per second or better frame rate required.
 - Better score is achieved when users do not move at all.
 - **Passive then active** – the engine first tries the passive liveness check, and if it fails, tries the active check. This mode requires colored images.
 - **Simple** – the engine requires user to turn head from side to side while looking at camera.
 - 5 frames per second or better frame rate recommended.
 - This mode can work with both colored and grayscale images.



Supported Cameras

These cameras are supported by VeriLook SDK:

- Any **webcam** or camera that is accessible using:
 - **DirectShow** interface for Microsoft Windows platform
 - **GStreamer** interface for Linux platforms.
 - **QuickTime** interface for Mac platform.
- Any built-in **smartphone** or **tablet** camera that is supported by **iOS** or **Android** OS. The camera should have at least 0.3 MegaPixel (640 x 480 pixels) resolution.
- These advanced cameras are supported:
 - IrisGuard IG-AD100 – face & iris camera (Microsoft Windows only)
 - VistaFA2 / VistaFA2E / VistaEY2 face & iris cameras (Microsoft Windows only)
- These models of **still cameras** are supported:
 - Canon EOS family still cameras (Microsoft Windows only)
 - Nikon DSLR still cameras (Microsoft Windows only; a specific camera model should support video capture)
- A **video file** can be also used as a data source for applications based on VeriLook SDK.
- Integrators can also write **plug-ins to support their cameras** using the plug-in framework provided with the Device Manager from the VeriLook SDK.

Simultaneous capture from multiple cameras is possible.

High resolution cameras may be used with VeriLook, but more powerful technology like *MegaMatcher SDK* or *SentiVeillance SDK* is recommended for acceptable performance. These cameras are supported:

- Any **IP camera**, that supports **RTSP** (Real Time Streaming Protocol):
 - Only **RTP over UDP** is supported.
 - **H.264/MPEG-4 AVC** or **Motion JPEG** should be used for encoding the video stream.
- Also these specific models of high-resolution cameras are supported:
 - Axis M1114 camera (Microsoft Windows and Linux)
 - Basler BIP2-1600-25c-DN IP camera (Microsoft Windows and Linux)
 - Cisco 4500 IP camera (Microsoft Windows only)
 - Mobotix S14D and DualNight M12 IP cameras (Microsoft Windows and Linux)
 - PiXORD N606 camera (Microsoft Windows and Linux)
 - Prosilica GigE Vision camera (Microsoft Windows and Linux)
 - Sony SNC-CS50 camera (Microsoft Windows and Linux)



System requirements

There are specific requirements for each platform which will run VeriLook-based applications.

Microsoft Windows platform requirements

- Microsoft Windows **XP / Vista / 7 / 8 / 10**, 32-bit or 64-bit.
- 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 VeriLook algorithm. Please check if a particular processor model supports SSE2 instruction set.
- 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 face record) require about **50 MB of additional RAM**.
- **Free space on hard disk drive (HDD):**
 - at least 1 GB required for the development.
 - 100 MB required for VeriLook components deployment.
 - Additional space would be required in these cases:
 - VeriLook does not require the original face image to be stored for the matching; only the templates need to be stored. However, storing face images on hard drive for the potential future usage is recommended.
 - Usually a database engine runs on a separate computer (back-end server). However, DB engine can be installed on the same computer for standalone applications. In this case HDD space for templates storage must be available. For example, 10,000 templates (each with 1 face record) stored using a relational database would require about 50 MB of free HDD space. Also, the database engine itself requires HDD space for running. Please refer to HDD space requirements from the database engine providers.
- A **camera or webcam**. See the previous chapter for more details.
- **Database engine** or connection with it. VeriLook templates can be saved into any DB (including files) supporting binary data saving. VeriLook Extended SDK contains the following support modules for Matching Server on Microsoft Windows platform: **Microsoft SQL Server, MySQL, Oracle, PostgreSQL** and **SQLite**.
- **Network/LAN connection (TCP/IP)** for client/server applications. Also, network connection is required for using Matching server component (included in VeriLook Extended SDK). Communication with Matching server is not encrypted, therefore, 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 **.NET framework 3.5** or newer (for .NET components usage).
- One of following **development environments** for application development:
 - Microsoft Visual Studio 2008 SP1 or newer (for application development under C/C++, C#, Visual Basic .Net)
 - Sun Java 1.6 SDK or later.



Android platform requirements

- A smartphone or tablet that is running **Android 4.0 (API level 14)** OS or newer.
 - API level 19 is the recommended **target** for code compilation.
 - If you have a custom Android-based device or development board, contact us to find out if it is supported.
- ARM-based **1.5 GHz processor recommended** for face processing in the specified time. Slower processors may be also used, but the face processing will take longer time.
- At least **20 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, **1,000 templates** (each with 1 face record) require about **5 MB of additional RAM**.
- **Free storage** space (built-in flash or external memory card):
 - 30 MB required for embedded face components deployment for each separate application.
 - Additional space would be required if an application needs to store original face images. VeriLook does not require the original face image to be stored for the matching; only the templates need to be stored.
- Any smartphone's or tablet's **built-in camera** which is supported by Android OS. The camera should have at least 0.3 MegaPixel (640 x 480 pixels) resolution.
- **Network/LAN connection (TCP/IP)** for client/server applications. Also, network connection is required for using Matching server component (included in VeriLook Extended SDK). Communication with Matching server is not encrypted, therefore, 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.
- **PC-side development** environment requirements:
 - Java SE JDK 6 (or higher)
 - Eclipse Indigo (3.7) IDE
 - Android development environment (at least API level 14 required)
 - Apache Maven 3.1.x or newer
 - Internet connection for activating VeriLook component licenses



iOS platform requirements

- One of the following devices, running **iOS 8.0** or newer:
 - **iPhone 5** or newer iPhone.
 - **iPad 2** or newer iPad, including iPad Mini and iPad Air models.
 - **iPod Touch 6th Generation** or newer iPod.
- At least **20 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, **1,000 templates** (each with 1 face record) require about **5 MB of additional RAM**.
- **Free storage space** (built-in flash or external memory card):
 - 30 MB required for embedded face components deployment for each separate application.
 - Additional space would be required if an application needs to store original face images. VeriLook does not require the original face image to be stored for the matching; only the templates need to be stored.
- **Network/LAN connection (TCP/IP)** for client/server applications. Also, network connection is required for using Matching server component (included in VeriLook Extended SDK). Communication with Matching server is not encrypted, therefore, 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.
- **Development environment** requirements:
 - a Mac running Mac OS X 10.10.x or newer.
 - Xcode 6.4 or newer.



Mac OS X platform requirements

- A Mac running **Mac OS X 10.7.x** or newer. 2 GHz or better processor is recommended.
- 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 face record) require about **50 MB of additional RAM**.
- **Free space on hard disk drive (HDD):**
 - at least 1 GB required for the development.
 - 100 MB required for VeriLook components deployment.
 - Additional space would be required in these cases:
 - VeriLook does not require the original face image to be stored for the matching; only the templates need to be stored. However, storing face images on hard drive for the potential future usage is recommended.
 - Usually a database engine runs on a separate computer (back-end server). However, DB engine can be installed on the same computer for standalone applications. In this case HDD space for templates storage must be available. For example, 10,000 templates (each with 1 face record) stored using a relational database would require about 40 MB of free HDD space. Also, the database engine itself requires HDD space for running. Please refer to HDD space requirements from the database engine providers.
- A **camera** or **webcam**. See the previous chapter for more details.
- **Database engine** or connection with it. VeriLook templates can be saved into any DB (including files) supporting binary data saving. VeriLook Extended SDK contains **SQLite** support modules for Matching Server on Mac OS X platform.
- **Network/LAN connection (TCP/IP)** for client/server applications. Also, network connection is required for using Matching server component (included in VeriLook Extended SDK). Communication with Matching server is not encrypted, therefore, 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.
- Specific requirements for **application development:**
 - XCode 4.3 or newer
 - wxWidgets 3.0.0 or newer libs and dev packages (to build and run SDK samples and applications based on them)
 - Qt 4.8 or newer libs, dev and qmake packages (to build and run SDK samples and applications based on them)
 - GNU Make 3.81 or newer (to build samples and tutorials development)
 - Sun Java 1.6 SDK or later.



Linux x86 / x86-64 platform requirements

- Linux 2.6 or newer kernel (32-bit or 64-bit) is required. **Linux 3.0 kernel** or newer is recommended.
- 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 VeriLook algorithm. Please check if a particular processor model supports SSE2 instruction set.
- 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 face record) require about **50 MB of additional RAM**.
- **Free space on hard disk drive (HDD):**
 - at least 1 GB required for the development.
 - 100 MB required for VeriLook components deployment.
 - Additional space would be required in these cases:
 - VeriLook does not require the original face image to be stored for the matching; only the templates need to be stored. However, storing face images on hard drive for the potential future usage is recommended.
 - Usually a database engine runs on a separate computer (back-end server). However, DB engine can be installed on the same computer for standalone applications. In this case HDD space for templates storage must be available. For example, 10,000 templates (each with 1 face record) stored using a relational database would require about 50 MB of free HDD space. Also, the database engine itself requires HDD space for running. Please refer to HDD space requirements from the database engine providers.
- A **camera** or **webcam**. See the previous chapter for more details.
- glibc 2.11.3 library or newer
- GStreamer 1.2.2 or newer with gst-plugin-base and gst-plugin-good is required for face capture using camera/webcam or rtsp video. GStreamer 1.4.x or newer is recommended.
- **Database engine** or connection with it. VeriLook templates can be saved into any DB (including files) supporting binary data saving. VeriLook Extended SDK contains **MySQL, Oracle, PostgreSQL** and **SQLite** support modules for Matching Server on Linux x86 / x86-64 platforms.
- **Network/LAN connection (TCP/IP)** for client/server applications. Also, network connection is required for using Matching server component (included in VeriLook Extended SDK). Communication with Matching server is not encrypted, therefore, 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.
- Specific requirements for **application development:**
 - wxWidgets 3.0.0 or newer libs and dev packages (to build and run SDK samples and applications based on them)
 - Qt 4.8 or newer libs, dev and qmake packages (to build and run SDK samples and applications based on them)
 - GCC-4.4.x or newer
 - GNU Make 3.81 or newer (to build samples and tutorials development)
 - Sun Java 1.6 SDK or later.
 - pkg-config-0.21 or newer (optional; only for Matching Server database support modules compilation)



ARM Linux platform requirements

We recommend to contact us and report the specifications of a target device to find out if it will be suitable for running VeriLook-based applications. There is a list of common requirements for ARM Linux platform:

- A device with ARM-based processor, running **Linux 3.2 kernel** or newer.
- ARM-based **1.5 GHz processor recommended** for face processing in the specified time. .
 - **ARMHF architecture (EABI 32-bit hard-float ARMv7)** is required.
 - Lower clock-rate processors may be also used, but the face processing will take longer time
- At least **20 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, **1,000 templates** (each with 1 face record) require about **5 MB of additional RAM**.
- **Free storage** space (built-in flash or external memory card):
 - 100 MB required for VeriLook components deployment.
 - Additional space would be required if an application needs to store original face images. VeriLook does not require the original face image to be stored for the matching; only the templates need to be stored.
- A **camera** or **webcam**. See the previous chapter for more details.
- glibc 2.13 library or newer
- libstdc++-v3 4.7.2 or newer.
- **Network/LAN connection (TCP/IP)** for client/server applications. Also, network connection is required for using Matching server component (included in VeriLook Extended SDK). Communication with Matching server is not encrypted, therefore, 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.
- **Development environment** requirements:
 - GCC-4.4.x or newer
 - GNU Make 3.81 or newer
 - JDK 1.6 or later



Technical Specifications

640 x 480 pixels is the recommended minimal image size for faces' detection. Face template extraction and matching is not dependent on the image size.

32 pixels is the **minimal distance between eyes** for a face on image or video stream to perform face template extraction reliably. **64 pixels or more recommended** for better template extraction results.

All face templates should be loaded into RAM before identification, thus the maximum face template database size is limited by the amount of available RAM.

VeriLook has certain tolerance to face posture that assures face enrollment convenience:

- head **roll** (tilt) – ± 180 degrees (configurable);
 ± 15 degrees default value is the fastest setting which is usually sufficient for most near-frontal face images.
- head **pitch** (nod) – ± 15 degrees from frontal position.
- head **yaw** (bobble) – ± 45 degrees from frontal position (configurable);
 ± 15 degrees default value is the fastest setting which is usually sufficient for most near-frontal face images.

Also, see above the list of recommendations and constraints for facial recognition.

VeriLook 9.0 face detection algorithm can run in maximal speed or maximal accuracy modes. The face detection times in the table below are provided for 640 x 480 pixels images as ranges, where the smallest time corresponds to the **maximal speed** mode, and the largest time – to the **maximal accuracy**. The head pitch tolerance in the table below is always $\pm 15^\circ$.

VeriLook 9.0 face detection algorithm performance for all faces in a frame on PC-based systems (milliseconds)		
Roll tolerance	Yaw tolerance	Intel Core i7-4771
$\pm 15^\circ$	$\pm 15^\circ$	6 - 11
$\pm 15^\circ$	$\pm 45^\circ$	10 - 23
$\pm 45^\circ$	$\pm 45^\circ$	23 - 53
$\pm 180^\circ$	$\pm 15^\circ$	53 - 132
$\pm 180^\circ$	$\pm 45^\circ$	130 - 330

VeriLook biometric template extraction and matching algorithm is designed to run on **multi-core processors** allowing to reach maximum possible performance on the used hardware. Face template extraction is performed after all faces are detected in a frame. The template extraction time does not depend on image size.

VeriLook 9.0 face engine specifications				
	Android-based platform ⁽¹⁾		PC-based platform ⁽²⁾	
	Embedded Face Extractor	Embedded Face Client	Face Extractor	Face Client
Template extraction components				
Template extraction time (seconds)	1.34	1.20	1.34	0.60
Template matching components	Embedded Face Matcher		Face Matcher	
Template matching speed (faces per second)	3,000		40,000	
Single face record size in a template (bytes)	4,028 or 5,066 or 7,128 (configurable)			

(1) Requires to be run on Android devices based on at least Snapdragon S4 system-on-chip with Krait 300 processor (4 cores, 1.51 GHz).

(2) Requires to be run on PC or laptop with at least Intel Core 2 Q9400 quad-core processor (2.67 GHz) to reach the specified performance.



Reliability Tests

We present the testing results to show the VeriLook 9.0 algorithm template matching reliability evaluations. The following **public datasets** were used:

- NIST Special Database 32 - Multiple Encounter Dataset (MEDS-II). - <http://www.nist.gov/itl/iad/ig/sd32.cfm>
 - All full-profile face images from the dataset were removed because they are not supported by VeriLook SDK. This resulted in 1,216 images of 518 persons.
- University of Massachusetts Labeled Faces in the Wild (LFW). - <http://vis-www.cs.umass.edu/lfw/>
 - According to the original protocol, only 6,000 pairs (3,000 genuine and 3,000 impostor) should be used to report the results. But recent algorithms are “very close to the maximum achievable by a perfect classifier” [http://people.cs.umass.edu/~elm/papers/LFW_survey.pdf]. Instead, as Neurotechnology algorithms were not trained on any image from this dataset, verification results on matching each pair of all 13,233 face images of 5,729 persons were chosen to be reported.
 - All identity mistakes, which had been mentioned on the LFW website, were fixed. Also, several not mentioned issues were fixed.
 - Some images from the LFW dataset contained multiple faces. The correct faces for assigned identities were chosen manually to solve these ambiguities.

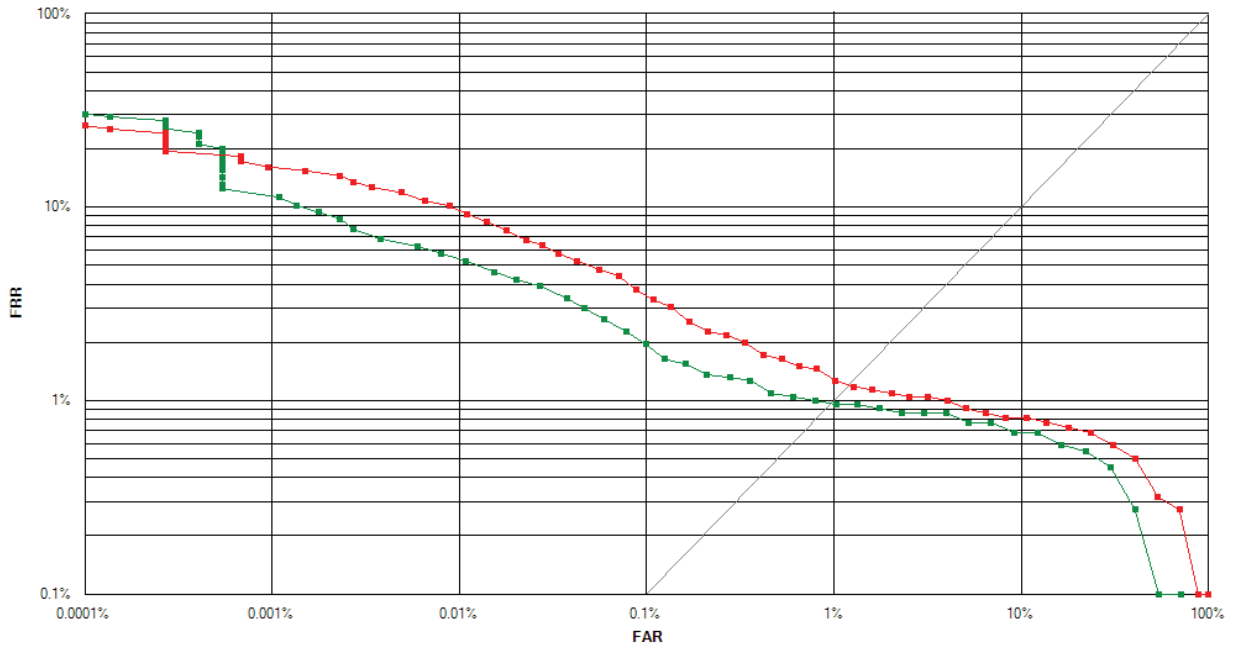
Both datasets contained faces, which are impossible to detect with the fastest near-frontal face detection. Face detection parameters were tuned to fully automatically detect maximum amount of faces with highest recall ratio using $\pm 45^\circ$ detectors, no speed optimizations, smaller search step and other parameters.

Two experiments were performed with each dataset:

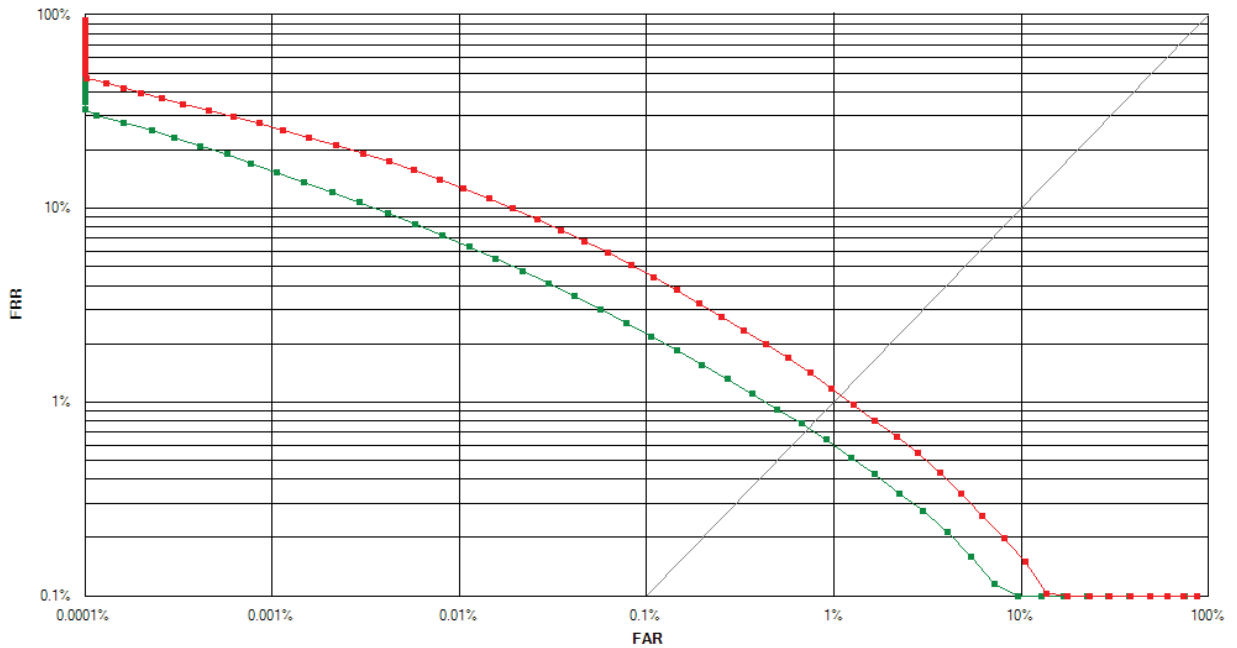
- **Experiment 1** maximized **matching accuracy**. VeriLook 9.0 algorithm reliability in this test is shown on the ROC charts as **green** curves.
- **Experiment 2** maximized **matching speed**. VeriLook 9.0 algorithm reliability in this test is shown on the ROC charts as **red** curves.

Receiver operation characteristic (**ROC**) curves are usually used to demonstrate the recognition quality of an algorithm. ROC curves show the dependence of false rejection rate (**FRR**) on the false acceptance rate (**FAR**). Equal error rate (**EER**) is the rate at which both FAR and FRR are equal. The ROC charts are available on the next page.

VeriLook 9.0 algorithm testing results with MEDS-II and LFW datasets				
	MEDS-II		LFW	
	Exp. 1	Exp. 2	Exp. 1	Exp. 2
Image count	1216		13233	
Subject count	518		5729	
Session count	1 - 18		1 - 530	
Image size (pixels)	variable		250 x 250	
Template size (bytes)	7128	5066	7128	5066
EER	0.8976 %	1.1440 %	0.7247 %	1.0740 %
FRR at 0.1 % FAR	1.9500 %	3.7190 %	2.5600 %	5.1120 %
FRR at 0.01 % FAR	5.7600 %	10.1600 %	7.2460 %	14.1100 %
FRR at 0.001 % FAR	12.4300 %	16.1500 %	17.0800 %	27.4800 %



VeriLook 9.0 SDK matching engine with face templates from NIST Multiple Encounter Dataset (MEDS-II):
■ Experiment 1, maximized matching accuracy scenario
■ Experiment 2, maximized matching speed scenario



VeriLook 9.0 SDK matching engine with face templates from the University of Massachusetts Labeled Faces in the Wild (LFW) dataset:
■ Experiment 1, maximized matching accuracy scenario
■ Experiment 2, maximized matching speed scenario



VeriLook Demo, Trial SDK and Related Products

VeriLook **algorithm demo** application and VeriLook **30-day SDK Trial** are available for downloading at www.neurotechnology.com/download.html.

These products are related to VeriLook SDK:

- **SentiVeillance SDK** - allows to create software for performing biometric face identification and pedestrian or vehicle or other object motion tracking using live video streams from high-resolution **digital surveillance cameras**. SentiVeillance component is capable of performing **automatic multiple face tracking, enrollment and matching** against database.
- **MegaMatcher SDK** – for development of AFIS or multi-biometric face, fingerprint, iris, voiceprint and palm print identification products.
- **MegaMatcher On Card SDK** – a product for fingerprint, iris and face matching on smart cards.
- **NCheck Bio Attendance** – an end-user employee attendance management application designed as **ready-to-use time and attendance system** with biometric fingerprint and face identification; the application uses VeriLook facial recognition algorithm to check person identity.
- **NVeiler Video Filter** – a plug-in for VirtualDub video processing application that is able to detect and hide faces in videos; VeriLook algorithm is used for the face detection feature of the plug-in.



Licensing VeriLook SDK

The following licensing model is intended for **end-user** product developers. Integrators who want to develop and sell a VeriLook-based development tool (with API, programming possibilities, programming samples, etc.), must obtain permission from Neurotechnology and sign a special VAR agreement.

Product Development

An integrator should obtain either a VeriLook 9.0 Standard SDK (EUR 339) or VeriLook 9.0 Extended SDK (EUR 859) to develop a product based on VeriLook technology. The SDK needs to be purchased just once and may be used by all the developers within the integrator's company.

VeriLook SDKs include a number of components; each particular component has specific functionality. A **license** for an individual VeriLook component is required for **each computer or device** that **runs** the component.

See the "Contents of VeriLook 9.0 Standard SDK and Extended SDK" chapter (the table on the page 4) for the list of component licenses included with VeriLook 9.0 Standard and VeriLook 9.0 Extended SDK.

Components are copy-protected – a license is required for a component to run. License activation options are listed below.

Additional component licenses may be obtained by VeriLook 9.0 SDK customers as required by their development process.

Product Deployment

To deploy a product developed with VeriLook 5.x / 9.0 SDK, an integrator need to obtain only the additional licenses required for the particular VeriLook 9.0 components that will run on **each computer or device** belonging to their customers. The available VeriLook components and license types for product deployment are the same as for product development.

Each VeriLook component running on a computer belonging to the integrator's customer requires a license. License activation options are listed below on this page.

Prices for VeriLook 9.0 SDK and additional VeriLook component licenses can be found in the next section.

Licensing Agreement

The Licensing Agreement (http://neurotechnology.com/mm_90_sla.html) contains all licensing terms and conditions.

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.



Single computer licenses

A single computer license allows the installation and running of a VeriLook component installation on one computer or device. Neurotechnology provides a way to renew the license if the computer undergoes changes due to technical maintenance.

Each single computer license requires **activation** for a VeriLook component to run. The available activation options are listed below.

Additional single computer licenses for VeriLook components may be obtained at any time by VeriLook SDK customers.

License activation options

Single computer and concurrent network licenses are supplied in three ways:

- **Serial numbers** are used to activate licenses for particular VeriLook components. The activation is done via the Internet or by email. After activation the network connection is not required for single computer license usage. Notes:
 1. Activation by serial number is **not suitable for iOS and ARM-Linux** platforms.
 2. Activation by serial number is **not suitable for virtual environments**.
- **Internet activation.** A special **license file** is stored on a computer or a mobile or embedded device; the license file allows to run particular VeriLook components on that computer or 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.
- 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

Volume license manager is used on site by integrators or end users to manage licenses for VeriLook components. It consists of license management software and a dongle, used to store the purchased licenses. An integrator or an end-user may use the volume license manager in the following ways:

- **Activating single computer licenses** – An installation license for a VeriLook component will be activated for use on a particular computer or an Android device. The number of available licenses in the license manager will be decreased by the number of activated licenses.
- **Managing single computer licenses via a LAN or the Internet** – The license manager allows the management of installation licenses for VeriLook components across multiple computers or Android 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.
- **Using license manager as a dongle** – A volume license manager containing at least one license for a VeriLook component may be used as a dongle, allowing the VeriLook component to run on the particular computer where the dongle is attached.

Additional VeriLook component licenses for the license manager may be purchased at any time. Neurotechnology will generate an update code and send it to you. Simply enter the code into the license manager to add the purchased licenses.

VeriLook 9.0 enterprise license

The VeriLook enterprise license allows an **unlimited use** of VeriLook components in end-user products for a specific territory, market segment or project. Specific restrictions would be included in the licensing agreement.

The enterprise license price depends on the application size and the number of potential users of the application within the designated territory, market segment or project.

For more information please contact us.



Prices for VeriLook products

- The prices are **effective June 1, 2016**. The prices may change in the future, so please **download and review the latest version** of the brochure before making an order.
- Quantity discounts do not accumulate over time, prices do not include local import duties or taxes.
- Customers with Solution Partner status are eligible for product discounts.

VeriLook SDK	
VeriLook 9.0 Standard SDK	€ 339.00
VeriLook 9.0 Extended SDK	€ 859.00

Face Verification component for all platforms (prices per single computer license)	
Quantity	Price
1 - 9	€ 1.75
10 - 19	€ 1.27
20 - 49	€ 1.13
50 - 99	€ 1.00
100 - 199	€ 0.89
200 - 499	€ 0.79
500 - 999	€ 0.69
1000 - 1999	€ 0.62
2000 and more	Please contact us for more information

Face components for PCs (prices per single computer license)					
Quantity	Face Extractor	Face Client ⁽¹⁾	Face Token Image ⁽¹⁾	Face BSS ⁽¹⁾	Face Matcher
1 - 9	€ 20.00	€ 60.00	€ 40.00	€ 10.00	€ 25.00
10 - 19	€ 15.00	€ 44.00	€ 30.00	€ 7.50	€ 18.00
20 - 49	€ 13.00	€ 39.00	€ 26.00	€ 6.50	€ 16.00
50 - 99	€ 11.00	€ 34.00	€ 23.00	€ 5.50	€ 14.00
100 - 199	€ 10.00	€ 30.00	€ 20.00	€ 5.00	€ 12.50
200 - 499	€ 9.00	€ 27.00	€ 18.00	€ 4.50	€ 11.00
500 - 999	€ 8.00	€ 24.00	€ 16.00	€ 4.00	€ 10.00
1000 - 1999	€ 7.00	€ 21.00	€ 14.00	€ 3.50	€ 9.00
2000 and more	Please contact us for more information				

Embedded face components (prices per single computer license)			
Quantity	Embedded Face Extractor	Embedded Face Client ⁽¹⁾	Embedded Face Matcher
1-9	€ 9.00	€ 27.00	€ 11.00
10-19	€ 6.50	€ 20.00	€ 8.00
20-49	€ 5.80	€ 18.00	€ 7.10
50-99	€ 5.10	€ 15.00	€ 6.20
100-199	€ 4.60	€ 13.50	€ 5.60
200-499	€ 4.10	€ 12.00	€ 5.00
500-999	€ 3.60	€ 10.50	€ 4.40
1000-1999	€ 3.20	€ 9.50	€ 3.90
2000 and more	Please contact us for more information		

License management	
Volume license manager	€ 16.00

(1) These components are not available for VeriLook Standard SDK customers.

VeriLook products can be ordered:

- online, at www.neurotechnology.com/cgi-bin/order.cgi
- via a local Neurotechnology distributor; the list of distributors is available at www.neurotechnology.com/distributors.html