



NEUROtechnology



Object recognition
for robotics and
computer vision

SentiSight SDK



SentiSight SDK

Object recognition for robotics and computer vision

Document updated on **April 24, 2012**

CONTENTS

Object learning and recognition processes	3
Algorithm capabilities and requirements	5
Contents of SentiSight 3.1 SDK	6
System requirements	7
Technical specifications	9
Reliability and performance tests	10
SentiSight SDK Trial, algorithm demo and related products	11
Licensing SentiSight SDK	12
Prices for SentiSight products	14

SentiSight is intended for developers who want to use computer vision-based object recognition in their applications. It enables manual and fully automatic object learning, and searching for learned objects in images from almost any camera, webcam, still picture or live video in an easy, yet versatile, way.

SentiSight is available as a software development kit that allows the development of object recognition systems for Microsoft Windows or Linux platforms.

- Reliable innovative algorithm, that is tolerant to appearance, object scale, rotation and pose.
- Accurate object detection, processing and tracking in real-time.
- Webcams or other low cost cameras are suitable for obtaining object images.
- Available as multiplatform SDK that supports multiple programming languages.
- Reasonable prices, flexible licensing and free customer support.



Object Learning and Recognition Processes

SentiSight has two operation modes: learning and recognition. In learning mode, SentiSight algorithm creates object model by extraction object features from an image or video. In recognition mode, SentiSight finds and tracks the objects that match the previously stored object models.

Object Learning Process

In order to recognize an object in an image, the appearance of an object must first be memorized. In the learning phase, SentiSight algorithms extract specific object features from a video stream or single image and save them into what is known as an **object's model**.

In many cases there is more information in a video or single image than just the object you want SentiSight to learn, like a background, other objects in the room or a hand holding the object. Therefore, to learn an object, information about the exact location of the object in the image should be provided.

SentiSight supports 2 methods of object learning: manual and automatic.

Manual object learning is suitable for most situations. A user must perform these steps for manual object learning in the SentiSight-based application:

1. Outline object's shape on an image by marking object's corner points to build a polygon. The image can be provided by user from image file, video file or live video stream.
2. Choose if to use local features-based, or shapes-based algorithm, or both algorithms.
3. Optionally choose more images of the object and repeat Step 1 for each image. The algorithm assists the user by estimating an approximate shape of the object if the object in an image is recognized using data from previous images. Learning the object from different sides and angles results in better recognition quality.
4. Input the learned object name (ID) into the system.

Automatic object learning is suitable for lightweight movable objects. This learning procedure is based on detecting an object by excluding a static background and object's holder (usually a hand) from an image. **Fixed camera** is highly recommended to be used.

A user must perform these steps for automatic object learning in the SentiSight-based application:

1. Choose a background and direct the camera to it.
2. Choose a holder – an object that will be used to hold and move the learned object. A user's hand can be the "holder".
3. This "holder" should be presented it to the camera first, in various poses and configurations (if it is not rigid object) so that it can be learned by SentiSight.
4. Choose if to use local features-based, or shapes-based algorithm, or both algorithms.
5. After the holder has been learned, SentiSight is ready to learn the object itself, by having the holder rotate and move the object closer and further from the camera.
6. Input the learned object name (ID) into the system.

Therefore, the automatic method requires to use live video or to provide separate videos or image sets of background, holder and object. Also, the other background elements could be learned together with the object if the object is hardly separable from the background. This can affect the ability of the algorithm to recognize the unique qualities of the object and may result in the object being misclassified with other objects that have the same background.

Manual object learning should be used for objects that cannot be moved or if there are no way to provide separate media with objects background and/or holder. Thus, automatic learning provides less amount of user interaction with the system, but it is not as precise as manual learning. Also manual learning is suitable for wider range of cases.



Object Recognition Process

Object recognition requires no user interaction apart from providing a video file with the object or pointing a camera to the scene where the learned object is presented or will appear. When the object appears in the vision field, SentiSight tries to recognize it. If the object is recognized by SentiSight, object's name (ID) and coordinates are returned.

The SentiSight algorithm creates a model with possible views from different sides, in different 3D poses and in different lighting conditions in object learning stage. This object's model improves recognition capability.



Algorithm Capabilities and Requirements

All performance tests were made on Intel Core i7 processor with 4 cores running at 2.67 GHz.

SentiSight is designed to be as universal as possible and is able to perform fully automatic and manual object learning. The technology can be used for a wide range of tasks, including:

- Recognition of documents, stamps, labels, packaging and other items for sorting, logo masking, usage monitoring and similar applications
- Object counting and inspection for assembly lines and other industrial applications
- Augmented and extended reality applications for toys, games, device and Web applications such as: smart toys for children that recognize cards, images, pictograms, etc.; recognition of places based on photographs and recognition of products such as beverages, foods and other consumer goods.
- Robotic vision for navigation and manipulation
- Law enforcement applications for identification, such as tattoo recognition

The SentiSight 3.1 technology has these capabilities for advanced visual-based object learning and recognition:

- **Accurate object detection.** The SentiSight algorithm is able to find out:
 - **whether** a particular object is presented in a scene;
 - **where** the object is located in the scene;
 - **how many** instances of the object are there in the scene.
- **Two algorithms for object recognition.** Depending on the object type one of these algorithms (or both) may be used for successful recognition:
 - **Local features** based algorithm uses small details of an object as distinctive features that are extracted into an object model and are used later to recognize the object. This algorithm has fast performance but is not suitable for solid-colored, reflecting or transparent (glass etc.) objects.
 - **Shape** based algorithm is useful for the objects, which do not have any distinctive details but have stable external edges (boundaries) and / or internal edges. This algorithm has slower performance but allows to recognize most objects that are not recognized by the local features based algorithm.
- **Simultaneous multiple object recognition.** The SentiSight algorithm provides simultaneous multiple 2D and 3D object detection and recognition.
- **Object evaluation.** The algorithm is also able to estimate the region an object occupies in a scene, providing additional information about the size, orientation and scale of the recognized object.
- **Fast image processing.** SentiSight can process video streams in real time, so it can be used for real-time applications. Also the algorithm is able to run several threads on multi-core processors making the recognition several times faster.
- **Objects tracking mode.** The SentiSight 3.1 library has **multiple objects** tracking mode for tasks that need very fast image processing during object recognition stage. The tracking works with complex backgrounds and fast moving objects. Tracking is initialized if an object is recognized and located, then tracks the object until it changes somewhat in appearance, at which point tracking is reinitialized by recognition. In tracking mode SentiSight is able to process **more than 100 frames per second** (320 x 240 pixels, single object in a frame).



Contents of SentiSight 3.1 SDK

SentiSight 3.1 SDK is intended for developers who want to use computer vision-based object recognition in their applications. The SDK allows rapid development of computer vision-based object recognition systems using functions from the SentiSight library for **Microsoft Windows** or **Linux** platforms. Developers have complete control over SDK data input and output; therefore SDK functions can be used in connection with most cameras (including webcams), with any database and with any user interface.

SentiSight 3.1 SDK distribution package contains:

	Microsoft Windows	Linux
Components		
• SentiSight 3.1 installation license	1 single computer license	
• Device Manager library	+	+
Programming samples		
• C++	+	+
• C#	+	
• Visual Basic .NET	+	
Programming tutorials		
• C/C++	+	+
• C#	+	
• Visual Basic .NET	+	
• Sun Java 2	+	+
Documentation		
• SentiSight 3.1 SDK documentation	+	



System Requirements

- **PC with x86 (32-bit) or x86-64 (64-bit) processor:**
 - **SSE2 support is required.** Processors that do not support SSE2 cannot run the SentiSight 3.1 algorithm. Please check if a particular processor model supports SSE2 instruction set.
 - **SSSE3 support is recommended,** as the SentiSight 3.1 algorithm provides higher performance using this instruction set. Please check if a particular processor model supports SSSE3.
 - **64-bit architecture** allows to work with larger images, larger model databases and also increases general SentiSight 3.1 algorithm performance due to usage of 64-bit CPU registers.
- At least **256 MB** of free RAM should be available for the SentiSight-based application. Additional RAM may be required:
 - For applications that need to recognize objects using **large database** as the whole database must be loaded into RAM before recognition. The database size depends on objects quantity and number of templates saved in each object model. Each object model may be rather large due to using long videos for learning (as template learned from each frame is saved separately) and/or using multiple views for each object.
For example, a database of **100 object models** with **36 templates per model** will require about **25 MB** of RAM when local features recognition algorithm is used, or about **50 MB** when shape recognition algorithm is used (for 320 x 240 pixels resolution).
 - For applications that need to work with **high resolution videos.** Higher resolution allows to extract more features from objects, thus the object model sizes will be bigger. In general the template size has about linear dependence from the image or video resolution.
For example: a frame from **1 MegaPixel** camera is about **13 times larger** than a frame from a regular **320 x 240 pixels** (0.08 MegaPixel) camera. If an object model learned from a sequence of 320 x 240 frames occupies 250 KB of memory, then a model learned from the same sequence of frames obtained from 1 MegaPixel camera will require about 3.3 MB of memory. Note that each video frame is processed separately and the obtained object template is saved into the model.



- Optional camera or webcam. These cameras are supported by SentiSight:
 - Any **webcam** or camera that is accessible using:
 - **DirectShow** interface for Microsoft Windows platform
 - **GStreamer** interface for Linux platform
 - Also these specific models of **high-resolution** cameras are supported:
 - Axis M1114 camera (Microsoft Windows and Linux)
 - Cisco 4500 IP camera (Microsoft Windows only)
 - IrisGuard IG-AD100 – face & iris camera (Microsoft Windows only)
 - Mobotix DualNight M12 IP camera (Microsoft Windows and Linux)
 - PiXORD N606 camera (Microsoft Windows and Linux)
 - Prosilica GigE Vision camera (Microsoft Windows only)
 - Sony SNC-CS50 camera (Microsoft Windows and Linux)
 - VistaFA2 / VistaFA2E cameras (Microsoft Windows only)
 - VistaMT Multimodal Biometric Device (Microsoft Windows only)
- Microsoft Windows specific:
 - Microsoft Windows 2000/XP/2003/2008/Vista/7, 32-bit or 64-bit.
 - Microsoft .NET framework 2.0 or newer (for .NET components usage).
 - Microsoft DirectX 9.0 or later (for camera/webcam usage).
 - Microsoft Visual Studio 2005 SP1 or newer (for application development under C/C++, C#, Visual Basic .NET)
- Linux specific:
 - Linux 2.6 or newer kernel, 32-bit or 64-bit.
 - glibc 2.7 or newer
 - GStreamer 0.10.23 (with gst-plugin-base and gst-plugin-good) or newer (for object capture using camera/webcam)
 - udev-143 or newer with libudev (for camera usage)
 - GTK+ 2.10.x or newer libs and dev packages (to run SDK samples and applications based on them)
 - GCC-4.0.x or newer (for application development)
 - GNU Make 3.81 or newer (for application development)



Technical Specifications

All specifications are given for Intel Core i7 processor with 4 cores running at 2.67 GHz.

The specifications are given for SentiSight 3.1 local features recognition and shape recognition algorithms. These algorithms can be used separately depending on object type, or together.

The specifications are provided for 320 x 240 pixels images. These performance dependencies from image area are valid for the same images with different resolutions:

- **Local features** based algorithm has **linear dependence** for object learning and **linearithmic ($n \log n$)** dependence for object recognition.
- **Shape** based algorithm has **linearithmic ($n \log n$)** dependence for object learning and **quadratic dependence** for object recognition.

Object model size depends on how feature-rich is an object, and thus is individual for each object.

These conditions may alter algorithms performance:

- **Rotation and translation.** The algorithm is generally rotation and translation invariant in a plane perpendicular to the camera. Also the algorithm is invariant for rotations up to 10-15 degrees out of a plane perpendicular to the camera. Different views of an object can be added to a model to handle larger rotations.
- **Resolution and scale changes.** Scale (size in image) difference between object's model and object itself can be up to 2-3 times. Objects should contain enough details, and be large enough to be recognized.
- **Occlusions.** The algorithm is robust to occlusions as big as 50 % of the objects size if enough unique edges remain visible.
- **Lighting conditions** (illumination, shadows and reflectance).
 - Planar objects only have problems with reflectance.
 - 3D objects have problems with varying lighting conditions, but constant lighting conditions do not cause many problems.
- **Transparency.** In general transparent objects are difficult to recognize.
- **Rigidity.** The algorithm can recognize only rigid objects. At least significant part of the object should be rigid.

Object recognition algorithms can be run in more than one thread on multi-core processors allowing to increase object model matching speed. The table below provides object recognition speeds as a range, where the smaller number means recognition speed using 1 thread, while the larger number means recognition speed using 8 threads. Note, that the specified processor executes 2 threads per one processor core in parallel.

SentiSight 3.1 object recognition algorithms technical specifications		
	Local features recognition algorithm	Shape recognition algorithm
Static Background Extraction / Object mask separation	25 frames per second	
Learning: Processing of single objects' frame	0.02 seconds	0.58 seconds
Learning: Generalization time (for 100 frames of object)	0.4 seconds	Not applicable
Recognition speed ⁽¹⁾ (models per second)	16,000 - 60,000	1,800 - 4,500

(1) When object model contains one template. Object model may contain multiple templates (usually corresponding with different viewpoints), therefore the algorithm will compare an object against all templates in the model before returning the recognition result. Also, this recognition speed is reached with sufficiently big databases (2,000 images and more); with smaller databases the recognition is slower.



Reliability and Performance Tests

All tests were performed on Intel Core i7 processor with 4 cores running at 2.67 GHz.

SentiSight 3.1 algorithm was tested with a subset of Amsterdam Library of Object Images (ALOI).

- The subset contained **objects 1-100** from ALOI.
- Images with object viewpoint variations (**ALOI-VIEW** collection) were used.
36 images per object were used.

Local features and shape based algorithms from SentiSight 3.1 were tested separately.

SentiSight 3.1 performance was tested on these image resolutions:

- **768 x 576 pixels** – the original *full resolution* images from ALOI.
- **320 x 240 pixels** – obtained by resizing the 768 x 576 images before testing.

At 0.1% False Acceptance Rate (FAR), the recognition rate is from 70% to more than 99% depending on object structural appearance, transparency, etc. For objects with well defined internal structure, the recognition rate is 98% - 99% at 0.1% FAR.

SentiSight 3.1 algorithms tests				
	Local features recognition		Shape recognition	
	768 x 576	320 x 240	768 x 576	320 x 240
Average learning time for 1 image (seconds)	0.0791	0.0143	1.4718	0.2968
Average learning time for 1 object (36 images) (seconds)	2.8484	0.5148	52.9835	10.6865
Average recognition speed when 1 thread is running (templates per second)	4561	25435	106	3614
Average recognition speed when 8 threads are running (templates per second)	17455	96435	265	9035
Average object model size (kilobytes)	722.80	222.17	3258.69	489.53

The ALOI database used in these tests is available at <http://staff.science.uva.nl/~aloi/>



SentiSight SDK Trial, Algorithm Demo and Related Products

Sentisight **algorithm demo** applications for Microsoft Windows and Linux platforms, and SentiSight **30-day SDK Trial** are available for downloading at www.neurotechnology.com/download.html.

These products are related to SentiSight SDK:

- **SentiSight Embedded SDK** – a product for object recognition on **Android** smartphones, tablets and other mobile devices. Produces **object models** that are the **same** as in SentiSight SDK, thus can be also used for developing client-side mobile applications for systems with server-side based on SentiSight SDK. See “SentiSight Embedded SDK” brochure for more information.
- **NVeiler Video Filter** – a plug-in for VirtualDub video processing application that is able to detect and hide faces in videos; SentiSight algorithm is used for face and object tracking feature of the plug-in.



Licensing SentiSight SDK

The following licensing model is intended for **end-user** product developers. Integrators who want to develop and sell a SentiSight-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 a SentiSight 3.1 SDK (EUR 339) to develop a product based on SentiSight technology. The SDK needs to be purchased just once and may be used by all the developers within the integrator's company.

SentiSight 3.1 SDK includes SentiSight component. A **license** for an individual SentiSight component is required for **each CPU** that **runs** the component (a processor can have any number of cores).

One single computer license for the SentiSight component is included with SentiSight 3.1 SDK.

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

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

Product Deployment

To deploy a product developed with SentiSight SDK, an integrator need obtain only the additional licenses required for the SentiSight components that will run on **each CPU** of their customer's computers. The available license types for product deployment are the same as for product development.

Each SentiSight 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 SentiSight 3.1 SDK and additional SentiSight component licenses can be found in the next chapter.

Please also refer to SentiSight SDK Software License Agreement on Neurotechnology web site for all licensing terms and conditions.

Single computer licenses

A single computer license allows the installation and running of a SentiSight component installation on one CPU (a processor can have any number of cores). 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 SentiSight component to run. The available activation options are listed below on this page.

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



License activation options

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

- **Serial numbers** are used to activate licenses for SentiSight components. The activation is done via the Internet or by email. After activation the network connection is not required for single computer license usage. Note: activation by serial number is not suitable for virtual environments.
- 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 SentiSight 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 SentiSight 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.
- **Managing single computer via a LAN or the Internet** – The license manager allows the management of installation licenses for SentiSight components across multiple computers 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 on the network.
- **Using license manager as a dongle** – A volume license manager containing at least one license for a SentiSight component may be used as a dongle, allowing the SentiSight component to run on the particular computer where the dongle is attached.

Additional SentiSight 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.

SentiSight 3.1 enterprise license

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

For more information please contact us.



Prices for SentiSight products

- The prices are **effective from April 2, 2012**. 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.
- The prices do not include any taxes.
- Product shipping cost depends on delivery country
- Our customers can gain a discount for our products by getting the Solution Partner status.

SentiSight SDK

SentiSight 3.1 SDK	€ 339.00
--------------------	----------

SentiSight 3.1 installation licenses (prices per single computer license)

Quantity	Price
1-9	€ 45.00
10-19	€ 33.00
20-49	€ 29.00
50-99	€ 25.00
100-199	€ 23.00
200-499	€ 21.00
500-999	€ 19.00
1000-1999	€ 17.00
2000-3999	€ 15.00
4000-7999	€ 13.00
8000 and more	Please contact us for more information

License management

Volume license manager	€ 16.00
------------------------	---------

SentiSight enterprise license

SentiSight 3.1 enterprise license	Please contact us for more information
-----------------------------------	--

SentiSight 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