



# SentiVeillance 7.2 Faces Technology Algorithm Demo

Users's guide

## Introduction

### About SentiVeillance SDK

SentiVeillance SDK is designed for developing software that performs biometric face identification, detects moving pedestrians or vehicles or other objects and performs automatic license plate recognition using live video streams from digital surveillance cameras. This smart monitoring technology can be used for a variety of applications, including retail and commercial areas, entrance monitoring and counting, automated time-attendance systems, transportation, law enforcement and security systems.

SentiVeillance includes a robust, dynamic face model that can adapt to visual appearance changes as subjects move across the scene. Based on the biometric facial recognition engine from Neurotechnology, SentiVeillance provides fast face detection, improved face feature determination, gender determination and emotion detection, including detection of smiles, closed-eyes, open-mouths, clear and dark glasses. SentiVeillance matches face images against internal databases, such as authorized personnel or criminal watch lists, and the associated application can then lock or unlock a door, put a record into a time/attendance system or immediately trigger a report or alert for recognized (or un-recognized) faces.

The SDK is used for passive identification – when passers-by do not make any efforts to be recognized. List of possible uses includes law enforcement, security, attendance control, visitor counting, traffic monitoring and other commercial applications.

The SentiVeillance SDK allows to create applications for Microsoft Windows and Linux platforms.

- Real time pedestrians and vehicles tracking and classification.
- Biometric facial identification and matching against watchlist database.
- Automated license plate recognition (ALPR) for moving vehicles.
- Color, size and movement vector estimation for vehicles and other objects.
- Gender classification, age evaluation, facial expression and attributes detection.
- Automatic operation logs and reports events, adds new faces from video stream to watchlist.
- Large surveillance systems support with multiple cameras.
- Ready-to-use server for integration into video management systems (VMS) optionally available.

The SentiVeillance technology has these specific capabilities:

- **Real time performance.** SentiVeillance technology performs face recognition, pedestrian or vehicle classification and tracking in real time. The technology is designed to run on multi-core processors to achieve fast performance.
- **Three modalities** for surveillance systems. Depending on the surveillance system design, one of these modalities may be used:

- **Biometric face recognition** – based on deep neural networks and provides these capabilities for surveillance systems:
  - **Multiple face detection**, features extraction and template matching with the internal database in real time.
  - **Facial identification** reliability enables using large watchlist databases.
  - **Face tracking** is performed in all successive frames from the video source until they disappear from camera field of view. The face tracking algorithm uses dynamic face and motion prediction models that make it robust to occlusions like other objects or even other faces. The algorithm is able to continue tracking a face even when it re-appears after being fully covered by occlusions (like walls, furniture, posters etc).
  - **Gender** classification (optional) for each person in the frame.
  - **Age** determination (optional) for each person in the frame.
  - **Smile, open-mouth, closed-eyes, glasses, dark-glasses, beard and mustache** attributes detection (configurable).
- **Vehicle or human detection, classification and movement tracking** – performs object detection of moving and static objects in the scene, their classification and tracking until they disappear. These features are available for surveillance systems:
  - **Object classification**. SentiVeillance allows to perform object classification, locations and tracking based on its type. Currently these classes are available: Person, Car, Bus, Truck, Bike.
  - **Color estimation**. The algorithm returns most likely color estimation for cars and pedestrians. The estimated color values are: red, orange, yellow, green, blue, silver, white, black, brown, grey.
  - **Movement vector estimation**. The algorithm returns movement vector estimation values like: "north", "south", "south-east" etc.
  - **Tolerance to object visibility**. The detection algorithm works with partially visible objects and from great distance.
- **Automated license plate recognition (ALPR)** – once a vehicle has been detected, SentiVeillance ALPR algorithm detects and reads the license plate:
  - **Traffic data processing**. SentiVeillance algorithms can simultaneously read vehicle license plates from multiple moving vehicles.
  - **Tolerance to camera position**. Depending on camera resolution, the ALPR algorithm can read license plates from longer distance and higher angle.
  - **Preventing cheating with replaced license plates**. Integrators can use vehicle recognition and ALPR modalities together for making software logic which checks if

recognized license plate corresponds other registration data, like vehicle color or type, and not being spoofed or moved from another vehicle.

- **Automatic operation.** A system based on SentiVeillance 7.2 SDK is able to log on the fly all events. It can be configured to automatically report events like match with a watch list, or perform automatic enroll from video.
- **Large surveillance systems support.** SentiVeillance 7.2 SDK allows to integrate its technology into surveillance systems with multiple cameras. A common PC with a GPU can process multiple video streams simultaneously.
- **Video files processing.** SentiVeillance also accepts data from video files. The video files can be processed in real time as coming from a virtual camera or can be processed at maximum speed depending on hardware resources available.

This guide is dedicated to face tracking and recognition in digital surveillance cameras. The guide covers such topics as adding subjects to watch list (aka enrollment), face tracking and recognition, GUI of the demo program and some notes on parameters and other settings.

The Face modality from the SentiVeillance component performs real-time detection, biometric identification and tracking of all detected faces. The algorithms are able to track multiple faces simultaneously and match them with faces from internal database (i.e. a watch-list of suspects or a list of company employees). New faces may be enrolled to the database either automatically from video streams or manually as templates, which were created with VeriLook SDK. Person's gender, age or facial attributes can be optionally determined for each person in a frame.

The demo program can accept video streams from different sources like digital surveillance cameras, IP cameras, video files, USB cameras and etc.

---

*For more information about SentiVeillance SDK, download a free trial version of SDK from Neurotechnology [website](#) and read documentation in "Documentation" folder ("SentiVeillance SDK.pdf"). In this document you'll also find "Samples" section which explains how to use the same demo application with IP cameras.*

---

## System requirements and supported cameras

Recommended requirements for the system:

- **PC or server with x86-64 (64-bit) compatible processor:**
  - **3 GHz** or better processor with **6 cores** is recommended.
  - **AVX2 support is highly recommended.** Processors that do not support AVX2 will still run the SentiVeillance algorithms, but in a mode, which will not provide the specified

performance. Most modern processors support this instruction set, but please check if a particular processor model supports it.

- **Graphical processing unit (GPU):**
  - a GPU is **required in these cases:**
    - a SentiVeillance component with vehicle-human or license plate recognition modalities is run to process any number of video streams;
    - a SentiVeillance component with face modality is run to process more than 2 video streams.
  - **NVIDIA GeForce GTX 1080** GPU or better is recommended to process up to 10 video streams on a node.
  - **Compute Capability 3.5** or better should be supported by the GPU.
- At least **8 GB of RAM**.
- **High-resolution digital camera(s)**. Camera resolution may vary depending on the actual application. The recommended resolution is about 2 MegaPixel, as processing video from cameras with higher resolution will require more free RAM and more powerful processor to keep the acceptable frame rate. These supported cameras are suitable for using with SentiVeillance 7.2 SDK:
  - Any **IP camera**, that supports **RTSP** (Real Time Streaming Protocol):
    - Only **RTP over UDP** is supported.
    - **VLC** framework can be optionally used for reading video streams.
    - **H.264/MPEG-4 AVC** or **Motion JPEG** should be used for encoding the video stream.
  - Any high-resolution digital camera that is accessible using:
    - **DirectShow** or **Windows Media** or **Media Foundation** interfaces for Microsoft Windows platform;
    - **GStreamer** interface for Linux platform.
  - Any **other device support can be added by customers** using the provided Device Manager **Plug-in Framework**. Please refer to the SentiVeillance 7.2 SDK documentation for the detailed information.
- Microsoft Windows specific:
  - Microsoft Windows 7 / 8 / 10 / Server 2008 / Server 2008 R2 / Server 2012, 64-bit.

- Microsoft .NET framework 4.5 or newer (for .NET components usage).
- Microsoft DirectX 9.0 or later.
- Linux specific:
  - Ubuntu 16.04 OS
  - glibc 2.17 or newer
  - GStreamer 1.10.x or newer with gst-vaapi plugins installed for hardware accelerated video decoding
  - libgudev-1.0 219 or newer
  - wxWidgets 3.0.0 or newer libs and dev packages (to build and run SDK samples and applications based on them)
  - Sun Java 1.8 SDK or later (for application development with Java)

## Basic usage recommendations

Face recognition accuracy of SentiVeillance heavily depends on the quality of a face image in a frame. There are some basic recommendations and constraints when using face recognition applications based on SentiVeillance SDK.

### General recommendations:

- Image quality during enrollment is important, as it influences the quality of the face template. Enrollment from photo or video stream is possible.
  - Several images during enrollment are recommended for better facial template quality which results in improvement of recognition accuracy and reliability.
  - Additional enrollments may be needed when facial hair style changes, especially when beard or mustache is grown or shaved off.
- 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.
- 1 MegaPixel or better camera resolution is recommended for face enrollment and recognition. Make sure that native resolution is provided by a camera, as some cameras or webcams may scale up native images to higher resolution without image quality improvement.

### Face posture:

The SentiVeillance face recognition engine has certain tolerance to face posture:

- Head roll (tilt) –  $\pm 15$  degrees;
- Head pitch (nod) –  $\pm 15$  degrees from frontal position.
  - The head pitch tolerance can be increased up to  $\pm 25$  degrees if several views of the same face that covered different pitch angles were used during enrollment.
- Head yaw (bobble) –  $\pm 45$  degrees from frontal position (configurable).
  - $\pm 15$  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  $\pm 45$  degrees yaw range from frontal position.

## Using Algorithm Demo application

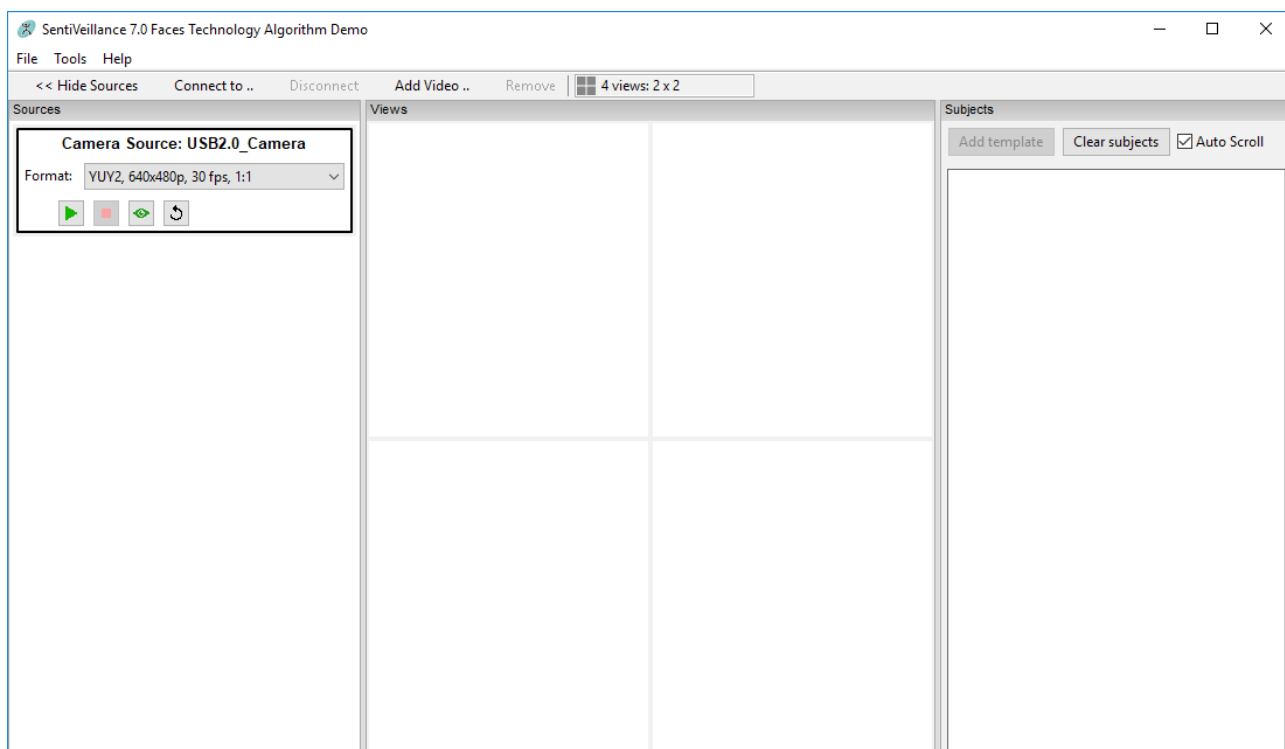
SentiVeillance demo application is used to demonstrate the capabilities of real time faces recognition and identification technology used in SentiVeillance SDK. This demo applications uses video streams from files or cameras. Using this application subjects can be added to internal database and after that identified in video streams.

SentiVeillance Faces Technology Algorithm Demo application is distributed as zip package for Windows and Linux OS (e.g., *SentiVeillance\_7\_0\_Algorithm\_Demo\_Win64\_x64\_2019-05-02.zip* or *SentiVeillance\_7\_0\_Algorithm\_Demo\_Linux\_x86\_64\_2019-05-02.zip*). Extract files form a package and start application (*SentiVeillanceFacesAlgorithmDemo.exe* for Windows or *SentiVeillanceFacesAlgorithmDemo* for Linux).

In this guide demo application for Windows is reviewed.

### Main screen

When you start an application, the main screen is displayed.



Let's review this screen:

- **Sources** – this view displays available video streams either from a camera or a video file.
- **Views** – videos from streams are displayed in this part of the screen. You can split this part into 1 – 10 views. Views are selected from a drop-down menu. Drag a video stream to a split view. It is recommended to select one view for each video stream (views count equal to sources count).

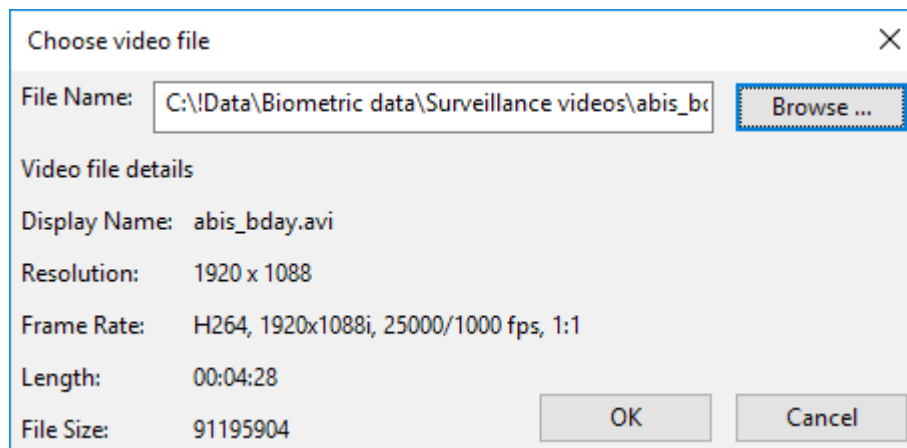


- *Subjects* – when a video stream is played and a subject is tracked, subject photo and subject details (time of appearance, subject Id, face details such as age, sex, glasses, beard and mustache) is displayed in this part of the window.
  - *Add template* – when you select a face image and press *Add template* button, a subject can be added to a watchlist. You need to specify subject Id of the located face.
  - *Clear subjects* – removes all tracked subjects from this view.
  - *Auto scroll* – is checked, this view scrolls down with an each new subject. Otherwise, user should scroll down.
- *Hide sources* – when you need more space on the screen, you can hide video sources (streams).
- *Connect to* – opens a window used to connect to a device (camera).
- *Disconnect* – disconnects a device.
- *Add video* – adds a video stream from a file.
- *Remove* – removes previously added video file.
- *File -> Enroll from image* – enables to enroll subject from image file.
- *File -> Enroll from directory* – enables to enroll multiple subjects from selected directory containing face image files.
- *Tools -> Settings* – opens SentiVeillance demo application settings window.
- *Tools -> Watchlist* – opens SentiVeillance watchlist, the window where you can review all enrolled subjects.
- *Tools -> Clear watchlist* – removes all subjects from the watchlist.
- *Help -> About* – displays information about this demo application. Also in this window NPluginManager, the tool for device plugins management, can be accessed.

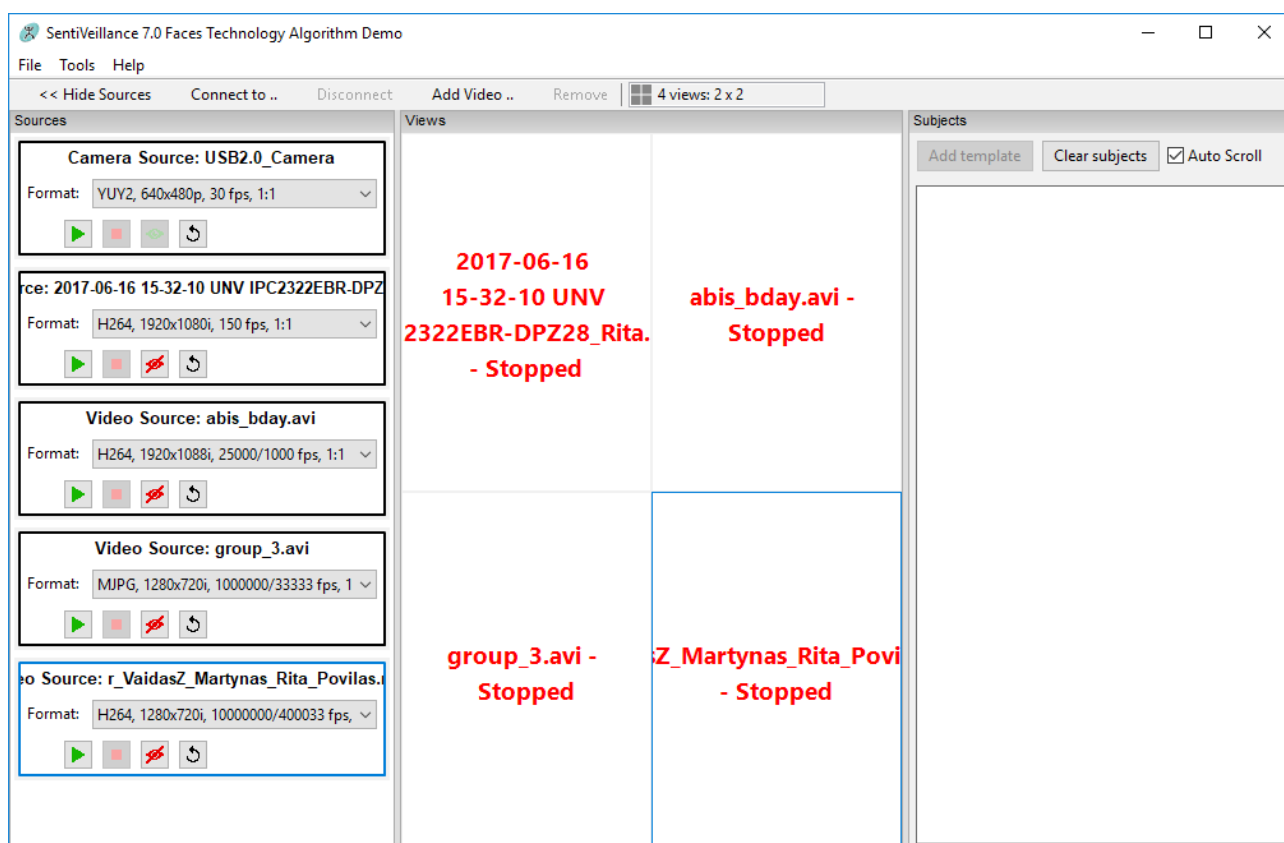
## Playing video stream sources

Let's start using application by adding 4 video files and playing them on the Views panel. For this you will need 4 different videos files, containing moving subjects.

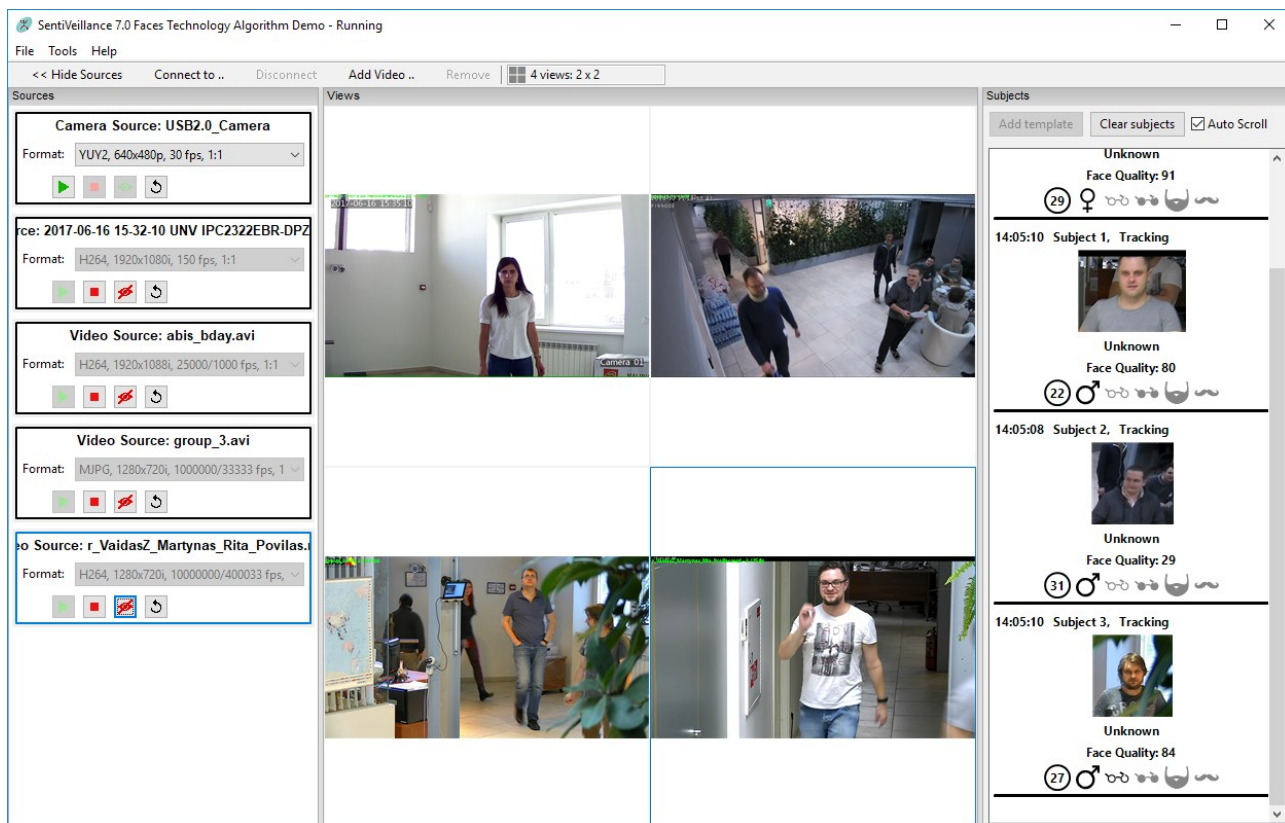
Select *Add video...* from the main screen and choose video file to import:



Added video files are displayed on the left. Select 4 views display and drag each video stream to a separate part of the *Views*. You should see similar display of the main screen:

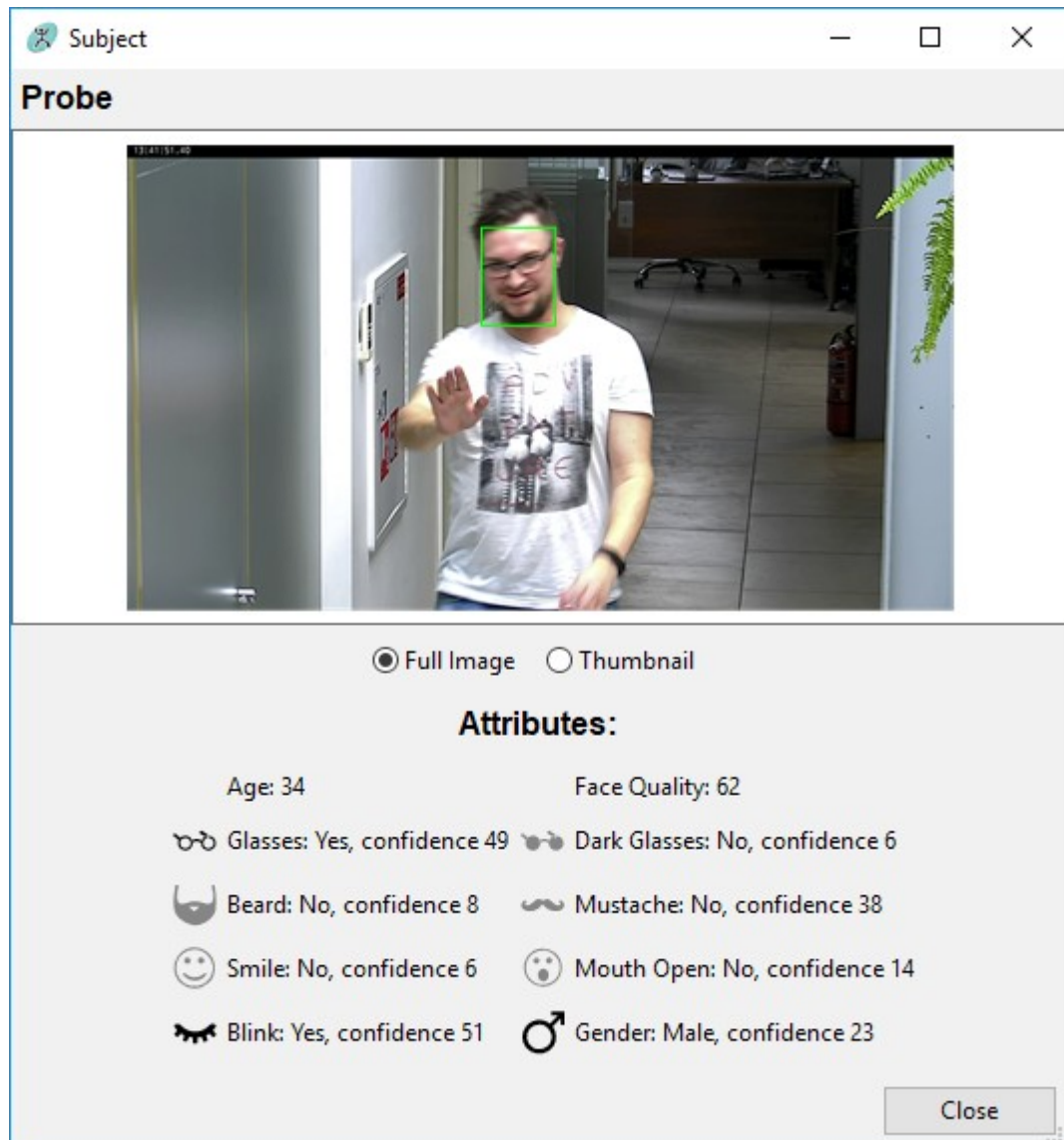


Now you can start each video stream by pressing *Play* button for each sources on the left. Tracked subjects are displayed on the right in the *Subjects* panel:



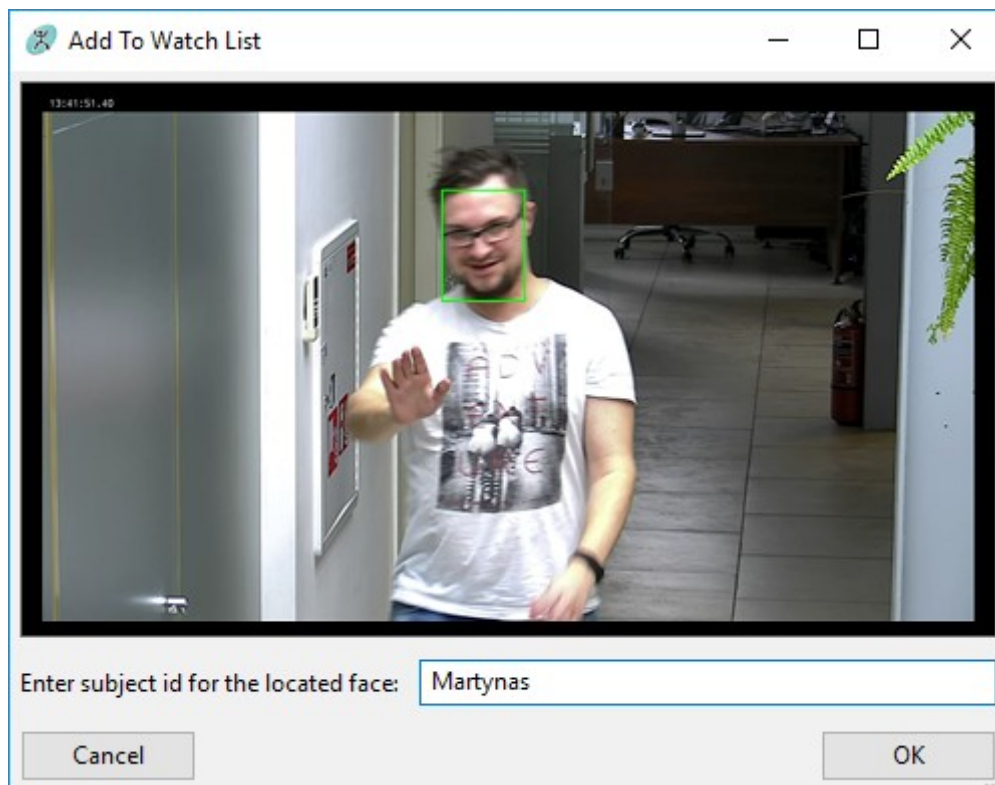
The SentiVeillance algorithm will generate a template of the subject's face by selecting best frames from the entire tracking sequence after subject disappeared.

Double click on a subject photo in the *Subjects* panel on the right. Subject probe with face attributes and their confidence values are displayed:



The bigger confidence value means it is more likely that face attribute was properly detected. In the previous screenshot glass confidence value is 49, which is quite a big value and we can see that the subject is wearing glasses. Contrary to glasses confidence value, beard confidence value is low – only 8. But as we can see from the screenshot, the subject has beard.

When you select a subject on the right panel and press *Add template*, a subject can be added to the watchlist (a subject can be added when he/she has disappeared). When this subject appears in a video stream and is tracked, subject Id is displayed on the *Subjects* panel.



16:19:43 Subject 66, Disappeared



Unknown

Face Quality: 80



16:19:40 Subject 67, Disappeared



Martynas (71)

Face Quality: 62



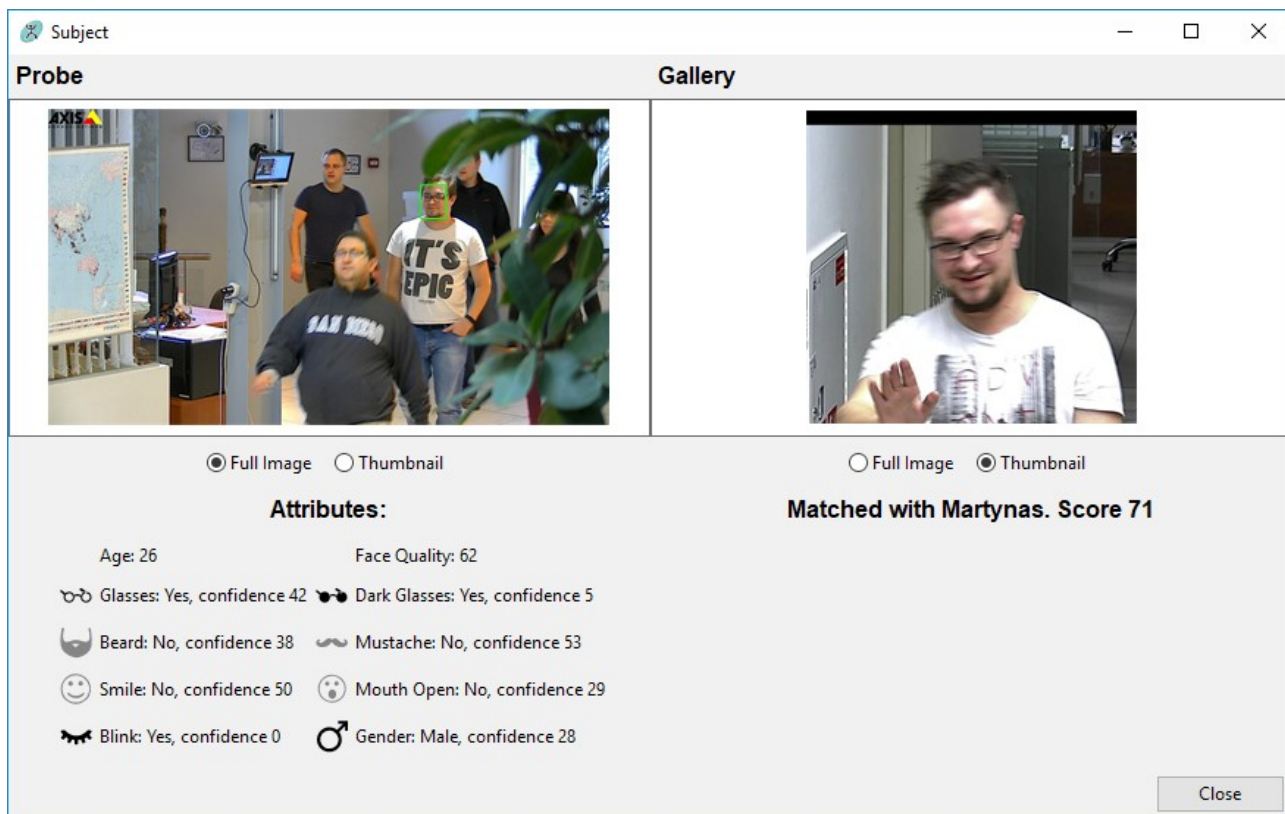
16:19:43 Subject 68, Disappeared



Unknown

Face Quality: 73

Matching result is displayed when you can double click on a matched subject. Face attributes, probe and watchlist images of a subject and matching score are displayed:



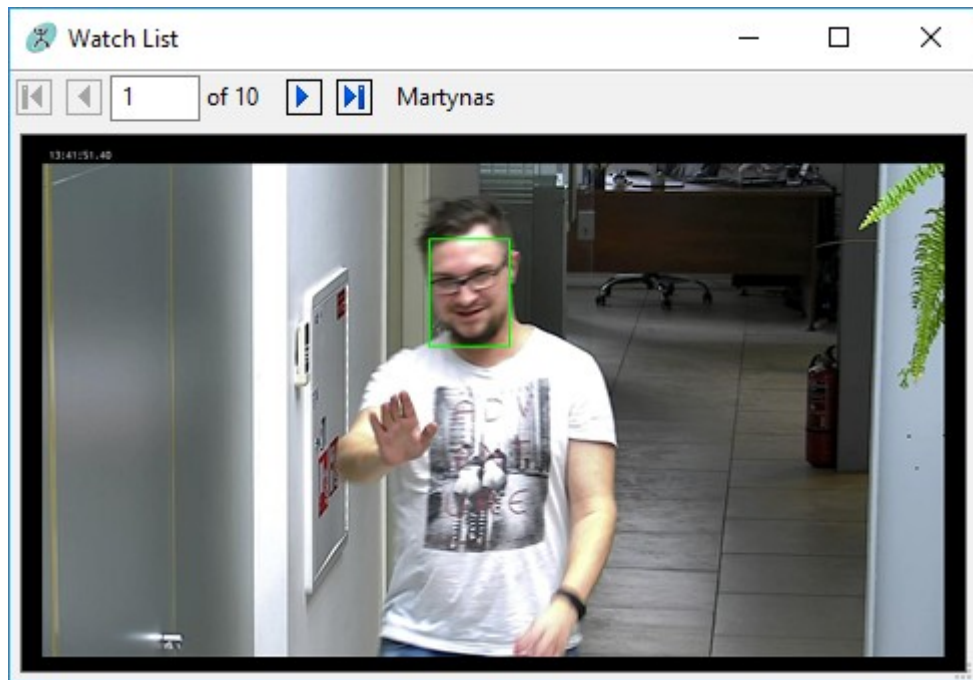
Theoretically, matching score can go up to infinity. But in the real life situations, depending on surveillance system setup and environmental conditions, the score of 70 and higher can be considered as a good matching result.

Subjects can be enrolled using images as well. To enroll from images go to *File* and select *Enroll from image(s)*. In the dialog window you will be able to select single or multiple images for enrollment. Enrollment from the directory can be done in the same way by selecting *Enroll from directory*.

**Warning.** During enrollment from images or a directory, templates will be generated for each image and automatically added to the Watch List. The template naming is done automatically according to the name of the image file.

You can always look at your Watch List by navigating to *Tools* and clicking Watch list button. The watch list shows the number of templates in it and names of templates. The displayed image is associated with a template and does not mean that it is saved in the template. This image is just for representative purposes. The real template uses information of one or more images obtained from face tracking results. Clearing Watch List is done by selecting *Clear watch list* from *Tools* menu.

Matching is done automatically as soon as *Watch List* has at least one record. After adding subjects to the *Watch List* you can select new video source and start surveillance. Algorithm will be detecting faces in video stream and tracking them, simultaneously performing matching of tracked faces in the *Watch List*.



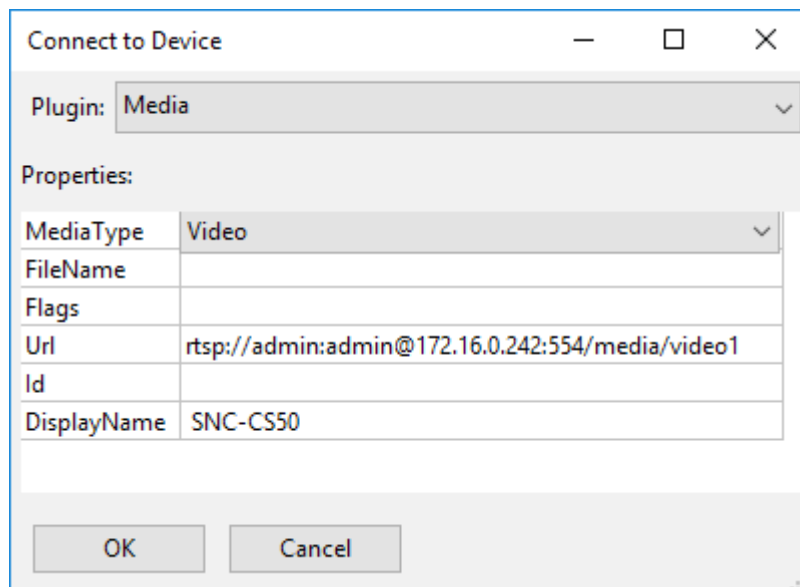
---

**Warning.** Clearing Watch List will clear the whole watch list. You cannot clear one particular record from the list (this is done by the demo program, SDK users can manage this according to their own needs).

---

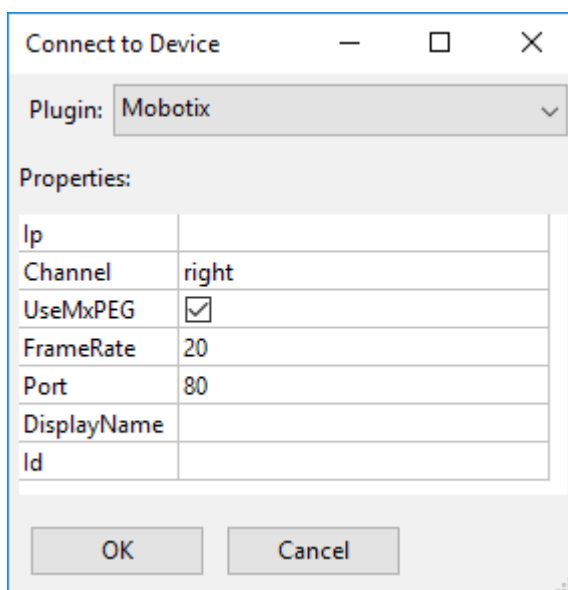
### Adding IP camera

While USB cameras are automatically added to *Sources* panel and can be used for video streaming, IP cameras should be added manually. Use *Connect to..* window and specify camera's IP address together with username and password. Note: camera should support RTSP protocol. In this example we are adding an IP camera which address is `rtsp://172.16.0.242:554/media/video1` and username and password are *admin*:



Added IP cameras are displayed in the *Sources* panel. You should drag it to the *Views* panel and start with the *Play* button.

Some IP cameras has plugins for simple connection and settings. You can select it from *Plugin* drop-down menu. This screenshot displays Mobotix cameras connection window (use camera's IP address in the same format as previously):

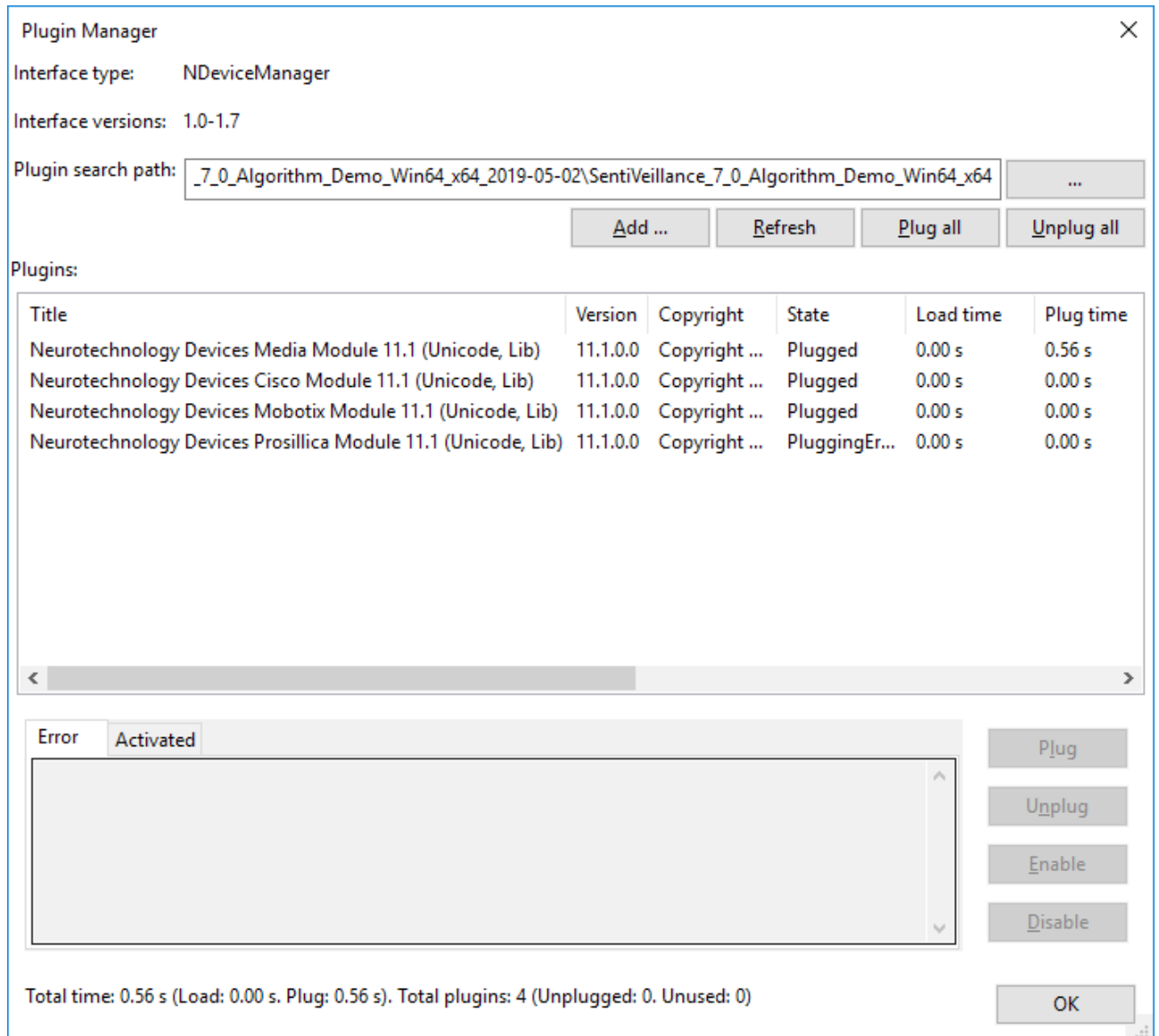


## Using plugin manager

Cameras plugins can be managed using the tool *NPluginManager*. It is accessed from SentiVeillance demo application About box (*Help* -> *About*) by selecting *NPluginManager* and clicking *Open* button.

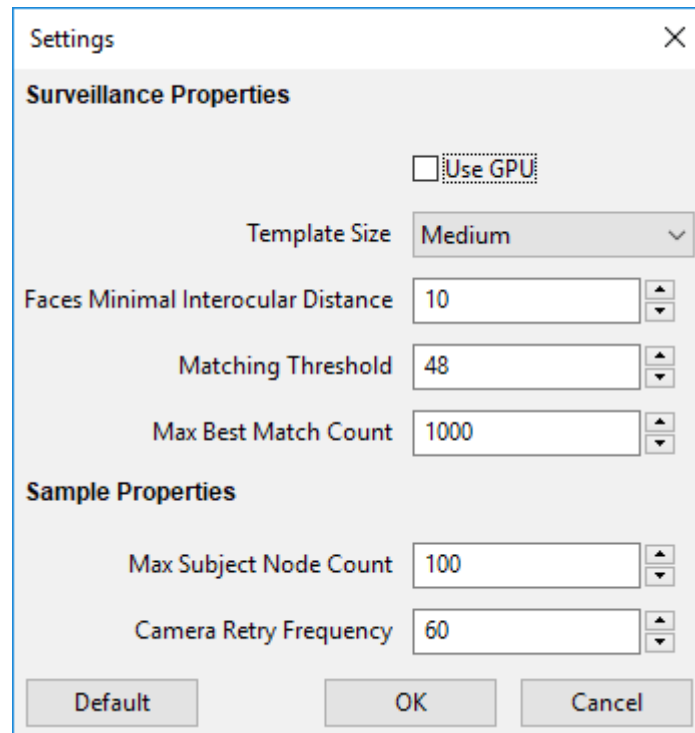


New plugins can be added by clicking *Add..* button. Plugins for cameras are saved in `\Cameras` directory. When you select an added plugin, with the help of buttons in this window you can plug or unplug and enable or disable them.



## Application Settings

Application settings can be accessed by choosing menu item *Tools->Settings*. This allows to change these settings:



- *Use GPU* – if GPU card is installed on your machine, use it for SentiVeillance processing speed acceleration. The GPU accelerates SentiVeillance demo application running on the CPU by offloading compute-intensive operations. You should note that when only CPU is used, this may negatively impact results. The best results are achieved with GPU unit.
- *Template size* – size of face image templates. Can be used Large, Medium or Small template. It is recommended to use medium template size. This parameter is used when generating a template from tracked subject.
- *Face minimal interocular distance* - this parameter sets the minimal distance between eyes for face detector. Value of 40 means that faces with less than 40 pixel between eyes will not be detected. The selection of this parameter depends on the camera setup. For low resolution cameras or cameras that capture faces from distance this parameter can be lower. Minimum value of this parameter is 8.
- *Matching threshold* – threshold that separates identical faces from different faces. Matching threshold is linked to false acceptance rate (FAR, different templates erroneously accepted as of the same) of matching algorithm. The higher is threshold, the lower is FAR and higher FRR (false rejection rate, same templates erroneously accepted as different) and vice a versa. The values may vary from 1 to infinity (theoretically). Lower values will give more false matches. The default value is set to 48, but in practice value of 70 or more would be a good choice.
- *Max best match count* – maximum number of matching results returned.
- *Max subject node count* - maximum number of items in *Subjects* node. Defines how many subjects should be displayed in the right panel. Default value is 100.

- *Camera retry frequency* – period of time in minutes after which application tries to reconnect to a camera.