VeriLook Surveillance Algorithm Demo 3.0

User's guide

Table of Contents

1 Introduction 1

1.1 System Requirements 1

2 Application 3

2.1 C++ (WxWidgets) 3

Index a

1 Introduction

VeriLook Surveillance Algorithm Demo 3.0 shows algorithm of VeriLook Surveiillance SDK which is a software development kit (SDK) for biometric face identification, pedestrian detection, object recognition and tracking using live video streams from high-resolution digital surveillance cameras. VeriLook Surveillance not only identifies faces in a crowd, it detects both people and objects while they are moving through the video frame, and it can detect if a person or object is entering or leaving a predefined boundary area, triggering an alert to the system monitor. 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.

VeriLook Surveillance 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, VeriLook Surveillance provides fast face detection, improved face feature determination and emotion detection, including detection of smiles, closed-eyes, open-mouths, clear glasses and dark glasses. VeriLook Surveillance 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.

In situations when video surveillance is monitoring a very large area or when cameras are in a position that does not allow some faces to be visible, VeriLook Surveillance uses moving object detection and classification to identify objects vs. people in the scene. In addition to person identification, it can be used to identify and monitor objects such as cars, buses and trucks in the camera’s view.

With smart monitoring technology, if a moving person or object has crossed a defined border or entered/left a specific monitored area or region, VeriLook Surveillance can send an immediate alert notification and trigger appropriate action if a restricted or sensitive area has been compromised. This ability to detect and track both moving people and objects allows VeriLook Surveillance to be used in a much wider range of applications, such as event security or police applications like the automatic detection of cars passing through red lights or stop signs.

VeriLook Surveillance continues tracking of subjects even when their faces briefly disappear from the frame or when they are partially blocked by other objects or even other faces (a common problem while tracking multiple faces). The face recognition algorithm enables detection of faces with up to 45 degrees out-of-plane rotation in yaw angle and uses motion prediction models to re-localize faces that have undergone full occlusion, such as when a subject has been fully obstructed by a wall and emerged on the other side. The dynamic face model allows the system to efficiently and reliably track faces in front of complex backgrounds and ensures that subjects can be localized in all video frames, even under strenuous conditions.

VeriLook Surveillance enables one computer to process images from multiple cameras and also provides connections between VeriLook Surveillance units deployed on different computers, synchronizing the databases so that the system works as a whole within the logical network. It enables real-time face detection, extraction and matching by providing embedded parallelization of all VeriLook functions for improved performance on multi-core, multi-processor systems.

1.1 System Requirements

Recommended requirements for system

- At least Intel Core 2 CPU with 4 cores running at 2.66 GHz or equivalent multi-core processor from other manufacturer
- At least 1 GB of free RAM
- Microsoft Windows specific requirements:
- Microsoft Windows XP / Vista / 7 / 8 / Server 2003 / Server 2008 / Server 2008 R2 / Server 2012
- Microsoft .NET framework 3.5 or later (for .NET components)
- Microsoft DirectX 9.0 or later

*** Note**: Windows 8 UI (Windows Store) applications are not supported.

- Linux specific requirements:
- Linux 2.6 or newer kernel, 32-bit or 64-bit. (based on glibc 2.7 or newer)
- GStreamer 0.10.23 (with gst-plugin-base and gst-plugin-good) or newer (for face 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)
- Sun Java 1.6 SDK or later (for application development with Java)
- wxWidgets 3.0.1 or newer libs and dev packages (to build and run SDK samples and applications based on them)
- libgudev-1.0 164-3 or newer (for webcam and microphone usage)
- One high resolution surveillance camera (about 1 Megapixel)

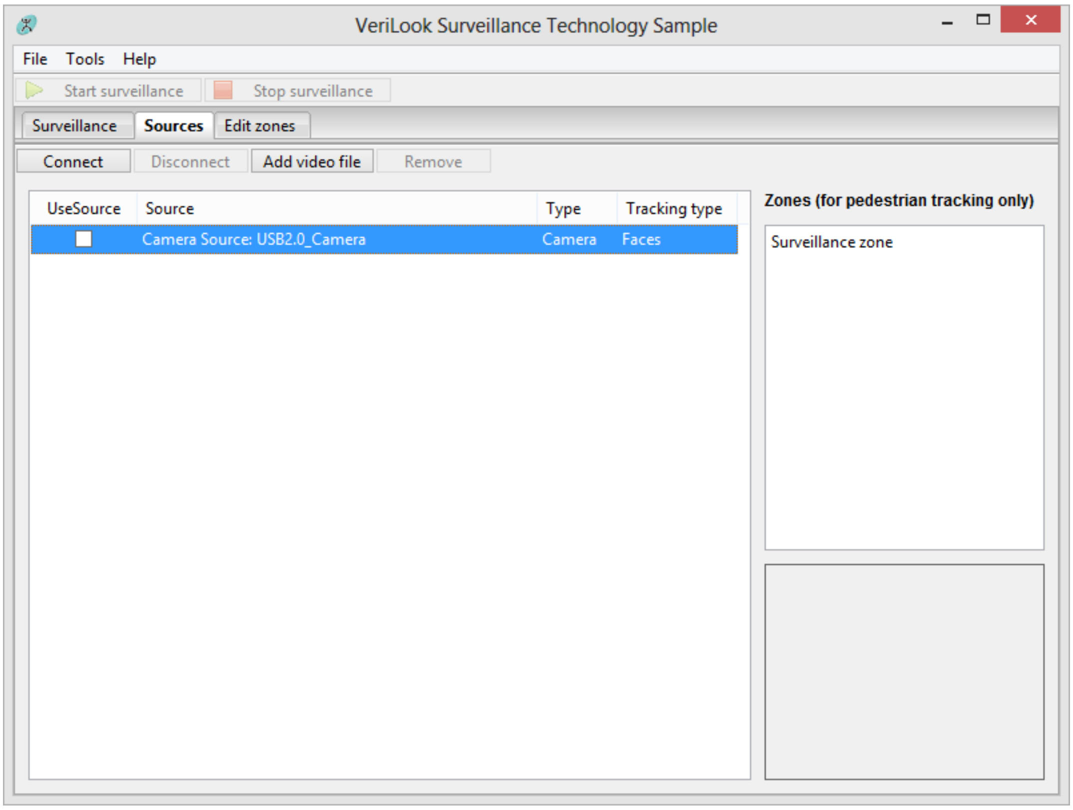
2 Application

VeriLook Surveillance Algorithm Demo 3.0 is written with C++ programming language and demonstrates algorithm of VeriLook Surveillance SDK.

2.1 C++ (WxWidgets)

C++ sample application is used for face identification, pedestrian detection, object recognition and tracking using live video streams (cameras or video files). Using this application subjects/objects can be added to internal database and after that tracked.

VeriLook Surveillance sample application can be started by launching `SurveillanceSampleWX.exe` file from `Bin\Win32_x86` or `Bin\Win64_x64` directories for Windows OS or `SurveillanceSampleWX` file from `Bin\Linux_x86` or `Bin\Linux_x86_64` directories for Linux OS. Source code for sample application is saved in `Samples\Surveillance\CPP\SurveillanceSampleWX` directory. After application was started, Main window is shown:



UseSource	Source	Type	Tracking type
□	Camera Source: USB2.0_Camera	Camera	Faces

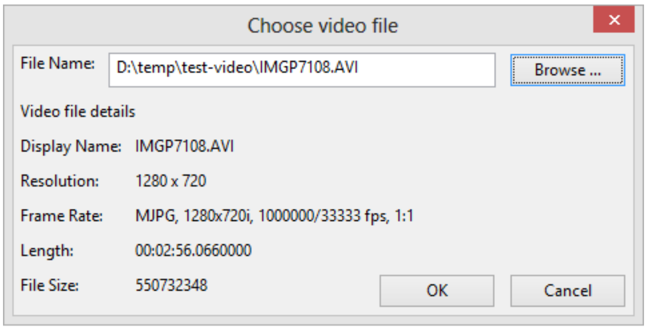
This window has 3 tabs:

- **Sources**. In this window video source (camera or video file) can be selected. Also zone for video can be assigned.
- **Surveillance**. In this window video is shown and tracking, recognition or identification is performed.
- **Edit Zones**. In this window new zone can be created or edited existing zone. Zone tracking is available for motion detection only.

When sample application is started, all installed cameras (web or IP cameras) are displayed in *Sources* tab. User can can select a camera and *Start surveillance* with live video stream from camera. But in this description we will show you how to work with video files. Using Surveillance sample with cameras is similar to using with video files.

Sources

Video file can be loaded by selecting *Add video file* in the *Sources* tab.

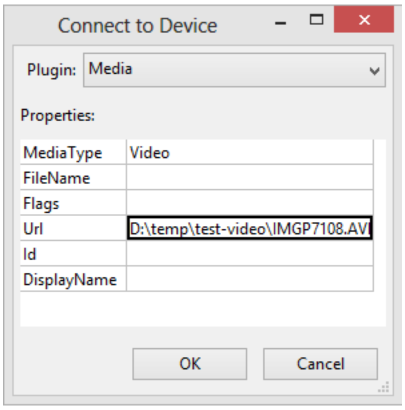


File Name: D:\temp\test-video\IMGP7108.AVI

Video file details

Display Name:	IMGP7108.AVI
Resolution:	1280 x 720
Frame Rate:	MJPG, 1280x720i, 1000000/33333 fps, 1:1
Length:	00:02:56.0660000
File Size:	550732348

Also video file can be loaded as Camera source. In this case you should select camera and press *Connect* button. *Connect to Device* window appears and you should select *Media* plugin and specify full path to video file as *Url*:

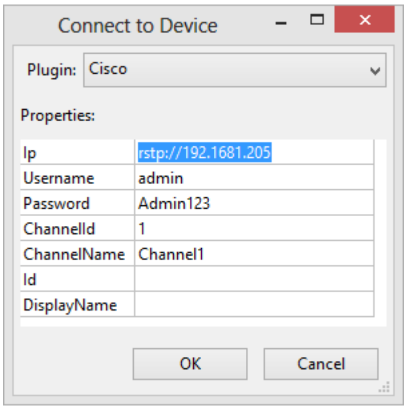


Plugin: Media

Properties:

MediaType	Video
FileName	
Flags	
Url	D:\temp\test-video\IMGP7108.AVI
Id	
DisplayName	

Connect to Device window also displays all available device plugins. If you need to connect to a network camera, you should select camera's plugin and setup connection parameters. Parameters for the Cisco network camera:

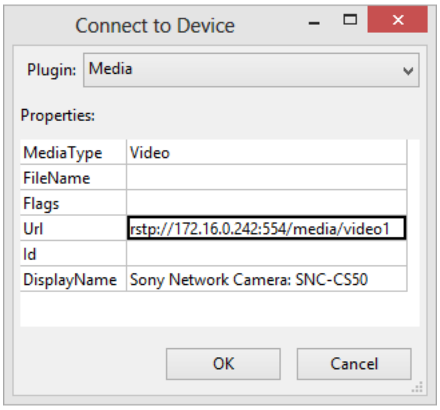


Plugin: Cisco

Properties:

Ip	rstp://192.1681.205
Username	admin
Password	Admin123
ChannelId	1
ChannelName	Channel1
Id	
DisplayName	

If your network camera does not have a plugin, you can connect to it choosing the *Media* plugin and specifying the camera's IP address. Also it is possible to set a display name for your camera:



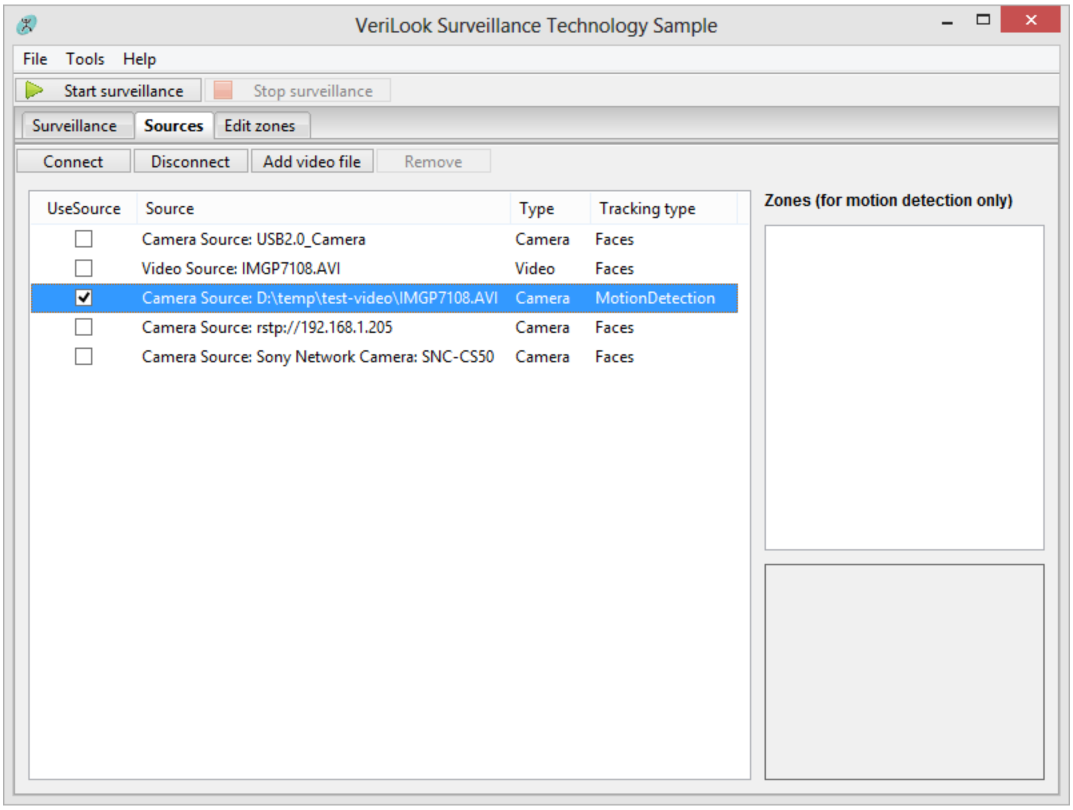
Plugin: Media

Properties:

MediaType	Video
FileName	
Flags	
Url	rstp://172.16.0.242:554/media/video1
Id	
DisplayName	Sony Network Camera: SNC-CS50

All these connection options are displayed in the *Sources.* The first option is an USB camera connected to a computer. The second option - loaded video file. The third - video file loaded as the camera source. The fourth - Cisco network camera. The fifth - the network camera with the specified display name.

As it was said previously in this sample documentation we will show how to work with video files. So lets choose video source (second or third option), tracking type (faces or motion detection) and press *Start surveillance* button.

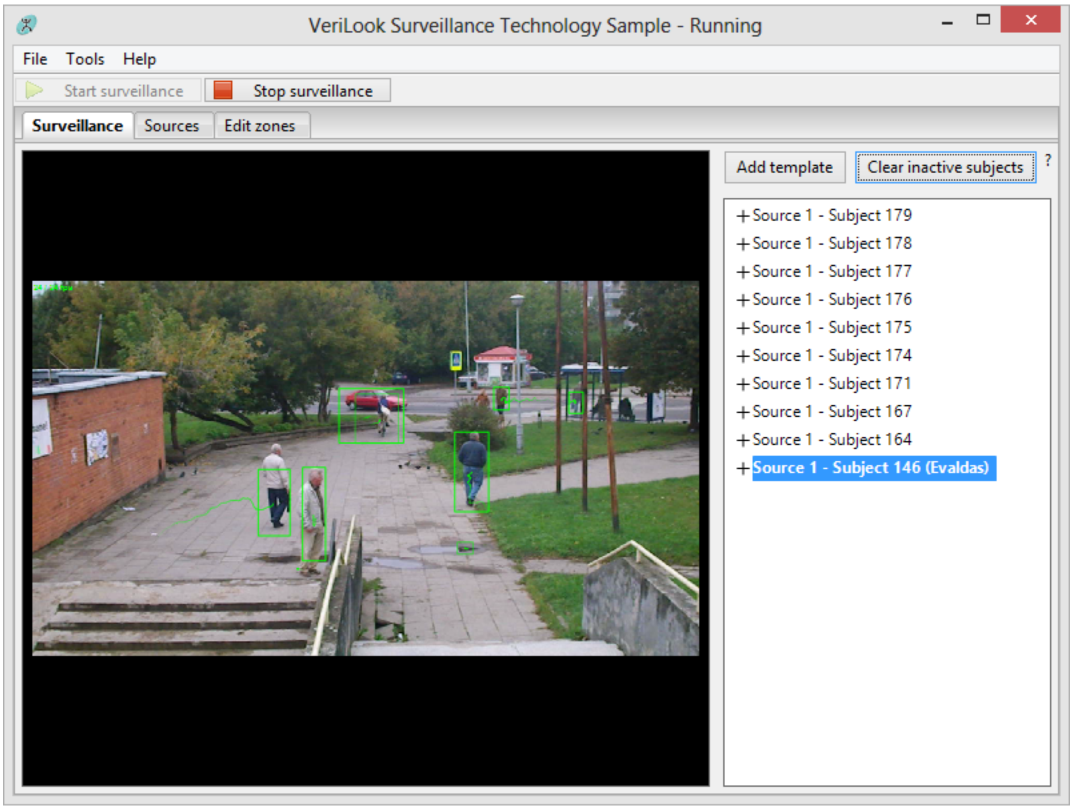


UseSource	Source	Type	Tracking type
☐	Camera Source: USB2.0_Camera	Camera	Faces
☐	Video Source: IMGP7108.AVI	Video	Faces
☑	Camera Source: D:\temp\test-video\IMGP7108.AVI	Camera	MotionDetection
☐	Camera Source: rstp://192.168.1.205	Camera	Faces
☐	Camera Source: Sony Network Camera: SNC-CS50	Camera	Faces

If tracking type is *MotionDetection*, all moving objects will be detected. If tracking type is *Faces*, only subject faces are tracked.

Surveillance

In this case we have chosen *MotionDetection* tracking type. Video is shown in *Surveillance* tab. When a moving object is tracked, a green rectangle with object tracking line is displayed:

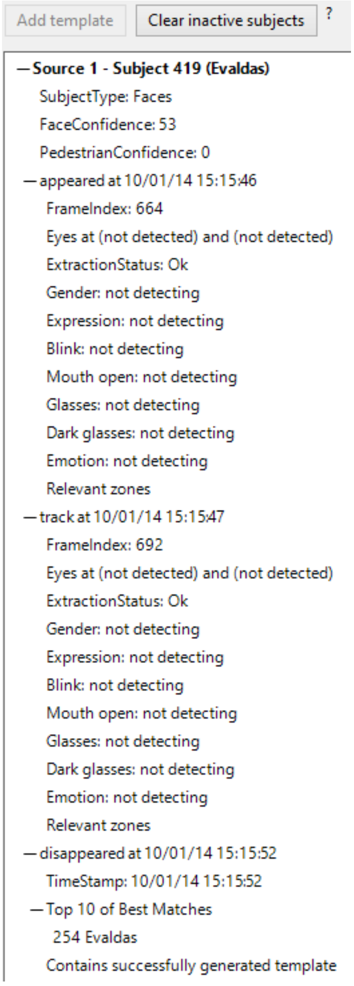


+Source 1 - Subject 179
+Source 1 - Subject 178
+Source 1 - Subject 177
+Source 1 - Subject 176
+Source 1 - Subject 175
+Source 1 - Subject 174
+Source 1 - Subject 171
+Source 1 - Subject 167
+Source 1 - Subject 164
+Source 1 - Subject 146 (Evaldas)

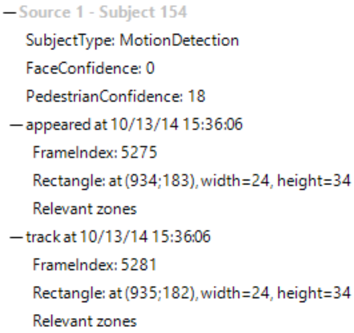
Tracked subjects are displayed in the right pane. If the subject was enrolled to database previously, his Id is shown in brackets. New subjects are enrolled to database by selecting list item (when it becomes bold) in the right panel and pressing button *Add template*. You should note that templates can be enrolled only when *Faces* tracking type is chosen.

Alternatively, new subjects can be enrolled from an image file or directory by clicking "File -> Enroll from image(s)" or "File -> Enroll from directory".

All list items (found subjects) in the right panel can be expanded. When list items are expanded, the time of subject's appearance and disappearance is shown for each subject.



```
— Source 1 - Subject 419 (Evaldas)
    SubjectType: Faces
    FaceConfidence: 53
    PedestrianConfidence: 0
  — appeared at 10/01/14 15:15:46
      FrameIndex: 664
      Eyes at (not detected) and (not detected)
      ExtractionStatus: Ok
      Gender: not detecting
      Expression: not detecting
      Blink: not detecting
      Mouth open: not detecting
      Glasses: not detecting
      Dark glasses: not detecting
      Emotion: not detecting
      Relevant zones
  — track at 10/01/14 15:15:47
      FrameIndex: 692
      Eyes at (not detected) and (not detected)
      ExtractionStatus: Ok
      Gender: not detecting
      Expression: not detecting
      Blink: not detecting
      Mouth open: not detecting
      Glasses: not detecting
      Dark glasses: not detecting
      Emotion: not detecting
      Relevant zones
  — disappeared at 10/01/14 15:15:52
      TimeStamp: 10/01/14 15:15:52
    — Top 10 of Best Matches
        254 Evaldas
      Contains successfully generated template
```



```
— Source 1 - Subject 154
    SubjectType: MotionDetection
    FaceConfidence: 0
    PedestrianConfidence: 18
  — appeared at 10/13/14 15:36:06
      FrameIndex: 5275
      Rectangle: at (934;183), width=24, height=34
      Relevant zones
  — track at 10/13/14 15:36:06
      FrameIndex: 5281
      Rectangle: at (935;182), width=24, height=34
      Relevant zones
```

The first picture shows an expanded list item when tracking type is *Faces*. In this case not only face appearance and disappearance time is shown but also face feature (such as gender, glasses, smile, etc.) status and Top 10 of best matches.

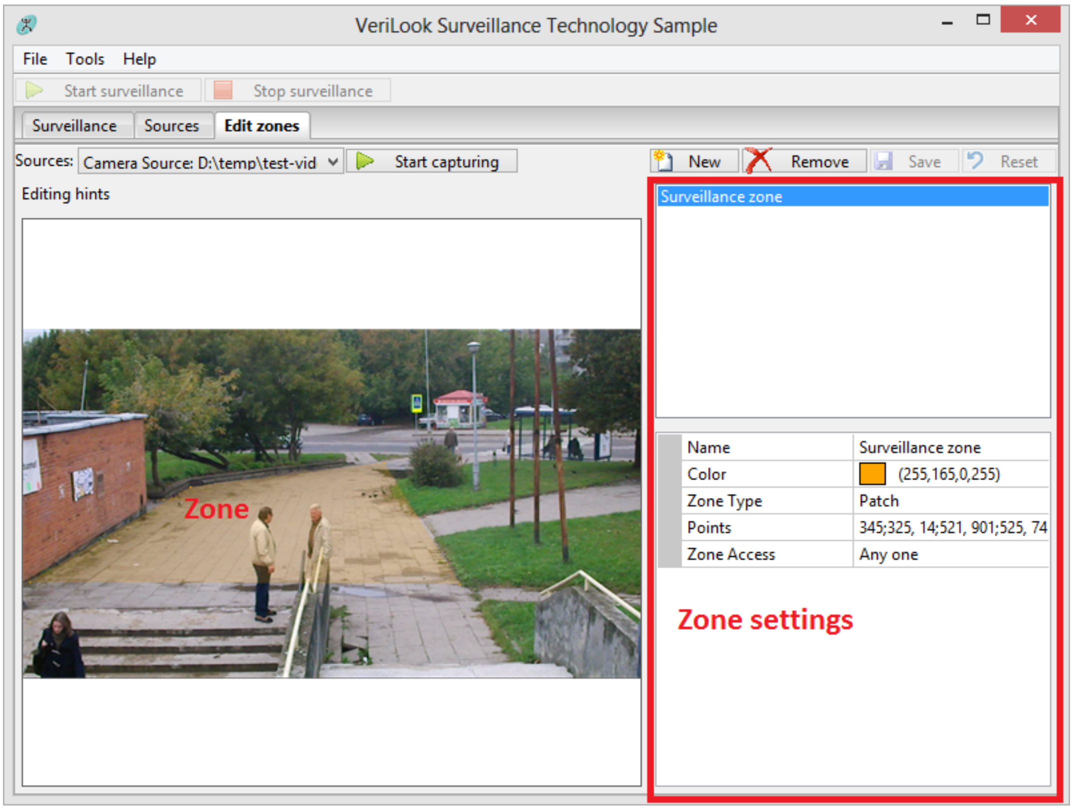
The second picture shows an expanded list item when tracking type is *MotionDetection*. In this list subject appearance and disappearance time are shown. Also face and pedestrian confidence, rectangle coordinates and Top 10 of Best Matches are shown.

Clear inactive subjects button can be used to clear inactive or disappeared subjects.

Edit zones

VeriLook Surveillance allows user to specify surveillance zone. A surveillance zone is a virtual polygon or a line, separating the camera's view. An event is an object crossing such a line, or entering/leaving a polygon. A zone involved in an event is referred to as a relevant zone.

Edit zones tab is used to create a new zone or edit an existing zone. To create a zone for an existing video or live stream from a camera select *Source* and press *Start capturing*. Video is shown on the left pane. The new zone is created by selecting *New* on the right pane. Then you can draw a polygon (should select *Patch* as zone type) or a tripwire (should select *Tripwire* as zone type) on the left pane. This will be the zone. A polygon or a tripwire is drawn by adding or removing points. This can be done by mouse click or manually adding point coordinates.



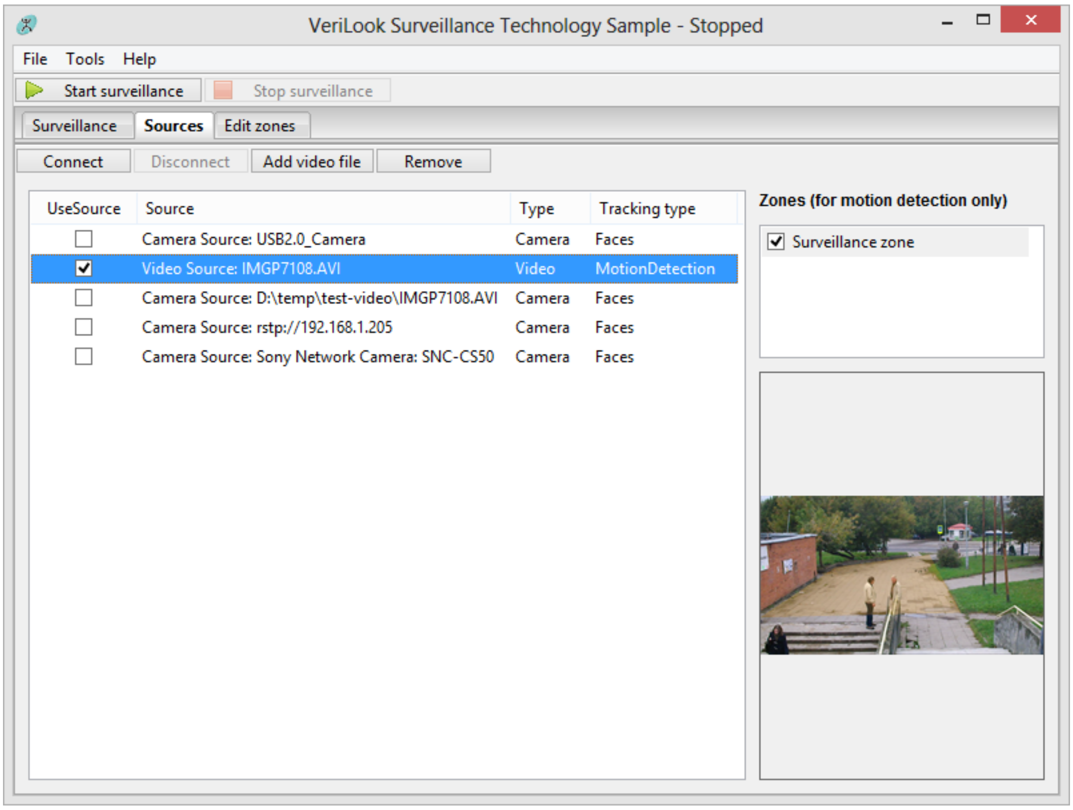
Name	Surveillance zone
Color	🟧 (255,165,0,255)
Zone Type	Patch
Points	345;325, 14;521, 901;525, 74
Zone Access	Any one

User can change a zone name, color, type and zone access on the right pane. Also zone point can be manually added or edited by selecting *Points*.

4 zone access types can be selected:

- Any one - any one can enter the zone. Tracked subjects will be displayed with a green rectangle.
- No one - no one can enter the zone. All tracked subjects in the zone will be displayed with a red rectangle.
- Allow cross forward - used for tripwire type zones and is used to display moving objects crossing zones in a wrong direction.
- Allow cross backwards - used for tripwire type zones and is used to display moving objects crossing zones in a wrong direction.

When all parameters are set, user should *Save* the zone. The zone is assigned for video source in *Sources* tab:

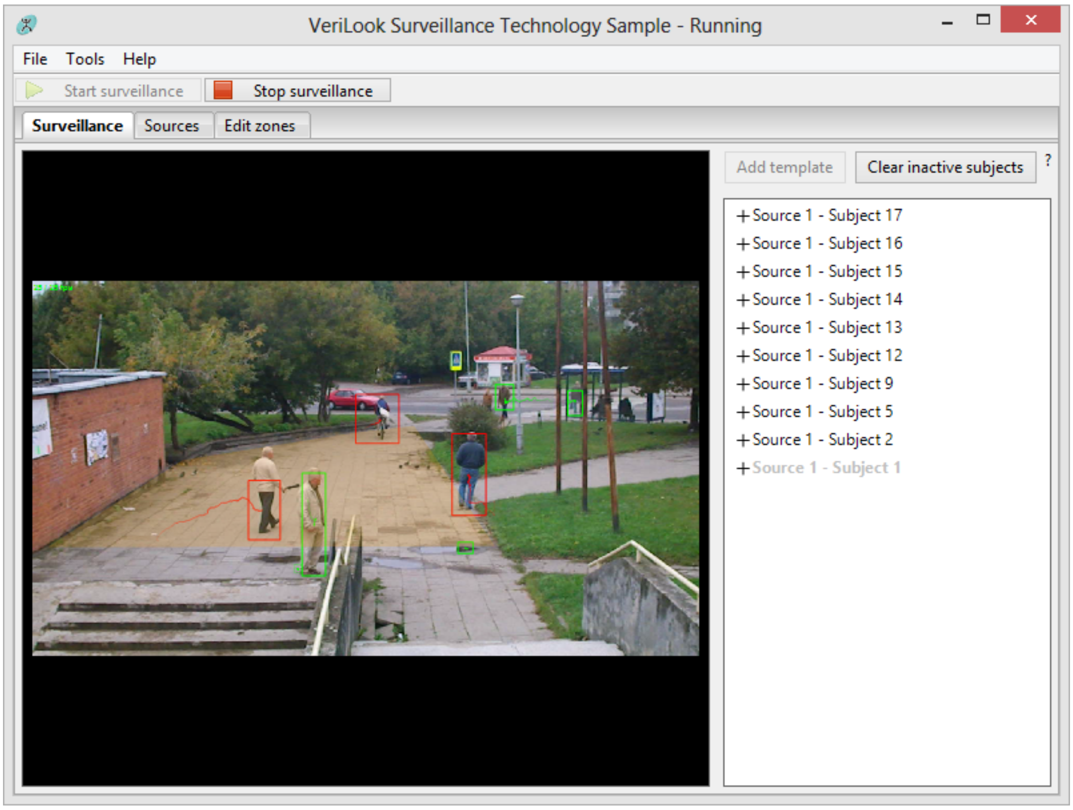


UseSource	Source	Type	Tracking type
□	Camera Source: USB2.0_Camera	Camera	Faces
☑	Video Source: IMGP7108.AVI	Video	MotionDetection
□	Camera Source: D:\temp\test-video\IMGP7108.AVI	Camera	Faces
□	Camera Source: rstp://192.168.1.205	Camera	Faces
□	Camera Source: Sony Network Camera: SNC-CS50	Camera	Faces

Zones (for motion detection only)

☑ Surveillance zone

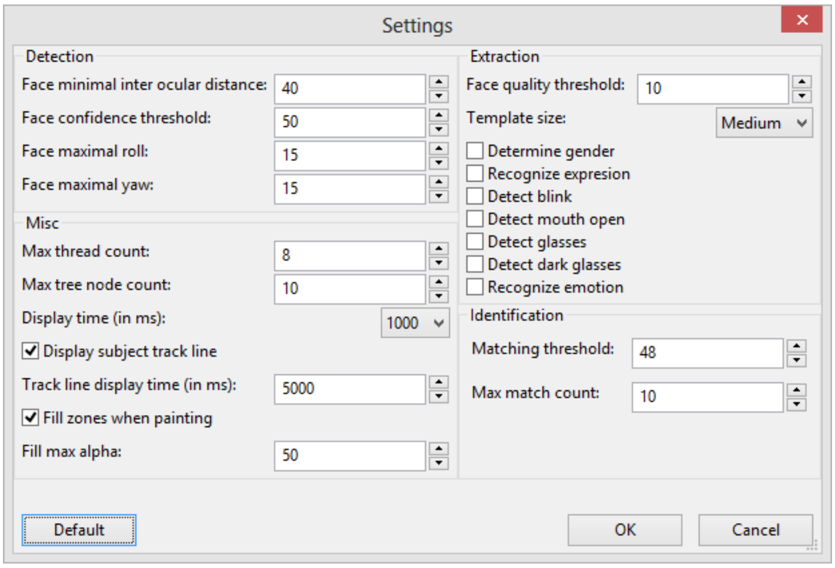
The picture below shows that moving objects (displayed with red rectangle) were tracked in the zone. Moving objects tracked outside the zone are displayed with green rectangle.



+ Source 1 - Subject 17
+ Source 1 - Subject 16
+ Source 1 - Subject 15
+ Source 1 - Subject 14
+ Source 1 - Subject 13
+ Source 1 - Subject 12
+ Source 1 - Subject 9
+ Source 1 - Subject 5
+ Source 1 - Subject 2
+ Source 1 - Subject 1

Application settings

Application settings can be accessed by choosing menu item *Tools->Settings*. This allows to change Detection, Identification, Extraction and other application settings:



Detection

Face minimal inter ocular distance: 40

Face confidence threshold: 50

Face maximal roll: 15

Face maximal yaw: 15

Misc

Max thread count: 8

Max tree node count: 10

Display time (in ms): 1000

☑ Display subject track line

Track line display time (in ms): 5000

☑ Fill zones when painting

Fill max alpha: 50

Extraction

Face quality threshold: 10

Template size: Medium

☐ Determine gender

☐ Recognize expresion

☐ Detect blink

☐ Detect mouth open

☐ Detect glasses

☐ Detect dark glasses

☐ Recognize emotion

Identification

Matching threshold: 48

Max match count: 10

Detection

- *Face minimal inter ocular distance* - minimum distance between eyes in a face. Faces which have smaller distance between eyes than this parameter, won't be returned by the face detection routines. Must be in range [10..10000]. The default value is 40 (See also NSP_DETECTION_MIN_IOD and DetectionMinIod).
- *Face confidence threshold* - specifies the threshold which is considered when looking for faces in an image. For each face candidate confidence parameter is calculated. With higher threshold value faces are selected more strictly by the face detection routines. Must be in range [0..100]. The default value is 50 (See also NSP_DETECTION_FACE_CONFIDENCE_THRESHOLD and DetectionFaceConfidenceThreshold).
- *Face maximal roll* - defines maximum roll angle deviation from frontal face in degrees which is considered when looking for faces in an image. Must be in range [0..180] (See also NSP_DETECTION_MAX_ROLL_ANGLE_DEVIATION and DetectionMaxRollAngleDeviation).
- *Face maximal yaw* - defines maximum roll angle deviation from frontal face in degrees which is considered when looking for faces in an image. Must be in range [0..180] (See also NSP_DETECTION_MAX_YAW_ANGLE_DEVIATION and DetectionMaxYawAngleDeviation).

Extraction

- *Face Quality Threshold* - specifies the threshold which is considered when extracting facial features from the image. With higher threshold better quality of face image is required to successfully extract facial features. The value of this parameter can be in range [0..255]. The default value is 10.
- *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.
- *Determine gender* - If checked, the application will detect gender (for this and below parameters see NleDetectionDetails)
- *Recognize expression* - recognize face expression (smile).
- *Detect blink* -detects blink (eyes closed).
- *Detect mouth open* - detects if mouth is open.
- *Detect glasses* - detects if face is with glasses.
- *Detect dark glasses* - detects if face is with dark glasses.
- *Recognize emotion* - if checked recognizes face emotion.

Identification

- *Matching threshold* - specifies matching threshold. Value should be more than 0.
- *Max match count* - -Maximum length of the list which is returned by each identification. The default value is 10.

Misc

- *Max thread count* - specifies the number of concurrent surveillance threads. Recommended value is "number of logical processors on CPU" plus one. Must be in range [3..8]. The default value is 8.
- *Max tree node count* - maximum number of items in tree node. Defines how many subjects should be displayed in the

right panel. Default value is 10.

- *Display Time (in ms)* - specifies the display time.
- *Display subject track line* - if checked, subject tracking line is shown on a screen.
- *Track line display time* - the maximum time for tracking line display.
- *Fill zones when painting* - if checked. zone will be filled with selected color.

Notes

When camera returns insufficient number of frames per second to sample application, green tracking rectangle is not drawn. In this case display time parameter can be used to set time for displaying tracking rectangle.

Index

A

Application 3

C

C++ (WxWidgets) 3

I

Introduction 1

S

System Requirements 1