



Detectors Pack

## Detector Pack. User Guide

Last update 02/07/2020

## Table of contents

<b>1</b>	<b>Introduction .....</b>	<b>6</b>
1.1	General information.....	6
1.2	The purpose of the document.....	6
1.3	Purpose of the Intellect detector pack .....	6
<b>2</b>	<b>Software and hardware requirements.....</b>	<b>7</b>
2.1	Platforms for the detection software modules operation.....	8
<b>3</b>	<b>Staffing requirements.....</b>	<b>9</b>
<b>4</b>	<b>Installing the Intellect Detector pack.....</b>	<b>10</b>
4.1	General information on installing the Intellect Detector pack.....	10
4.2	Installing the Intellect Detector pack .....	10
4.2.1	Description of the Intellect Detector pack installation files .....	10
4.2.2	Installation .....	10
4.2.3	Repair.....	13
4.2.4	Removal.....	15
4.2.5	Installing the Intellect Detector Pack in quiet (silent) mode .....	17
<b>5</b>	<b>Configuring detection modules.....</b>	<b>19</b>
5.1	General information on Neural analytics .....	19
5.2	Selecting a camera video stream for the detection modules.....	19
5.3	Configuring the events archive length .....	20
5.4	Queue length detection .....	20
5.4.1	Functionality of the Queue length detection module.....	20
5.4.2	Camera requirements for the Queue length detection module.....	20
5.4.3	Configuring the Queue length detection module .....	21
5.5	People counter detection .....	25
5.5.1	Functionality of the People counter detection module.....	25
5.5.2	Camera requirements for the People counter detection module.....	26
5.5.3	Configuring the People counter detection module .....	26
5.6	Stopped vehicle detection.....	31
5.6.1	Functionality of the Stopped vehicle detection module .....	31
5.6.2	Camera requirements for the Stopped vehicle detection module .....	31
5.6.3	Configuring the Stopped vehicle detection module .....	31
5.6.3.1	Licensing the Stopped vehicle detection module.....	31

5.6.3.2	Configuring the Stopped vehicle detection module .....	32
5.7	Glow detection .....	35
5.7.1	Functionality of the Glow detection module.....	35
5.7.2	Camera requirements for the Glow detection module.....	35
5.7.3	Configuring the Glow detection module .....	36
5.8	Heat map detection .....	39
5.8.1	Functionality of the Heat map detection.....	39
5.8.2	Camera requirements for the Heat map detection module .....	39
5.8.3	Configuring the Heat map detection module.....	40
5.9	Detection of moving against crowd flow .....	44
5.9.1	Functionality of the Detection of moving against crowd flow .....	44
5.9.2	Camera requirements for the Detection of moving against crowd flow module.....	44
5.9.3	Configuring the Detection of moving against crowd flow module .....	45
5.9.3.1	Behavior of the Detection of moving against crowd flow module .....	45
5.9.3.2	Configuring the Detection of moving against crowd flow module .....	45
5.10	Sweethearting detection .....	48
5.10.1	Functionality of the Sweethearting detection module.....	48
5.10.2	Camera requirements for the Sweethearting detection module .....	48
5.10.3	Configuring the Sweethearting detection module .....	48
5.11	Barcode detection.....	51
5.11.1	Functionality of the Barcode detection .....	51
5.11.2	Camera requirements for the Barcode detection module .....	51
5.11.3	Configuring the Barcode detection module .....	52
5.12	Train detection .....	54
5.12.1	Functionality of the Train detection module .....	54
5.12.2	Camera requirements for the Train detection module.....	54
5.12.3	Configuring the Train detection module .....	55
5.13	Fire detection and Smoke detection.....	58
5.13.1	Functionality of the Fire detection and Smoke detection modules.....	58
5.13.2	Camera requirements for the Fire detection and Smoke detection modules.....	58
5.13.3	Configuring the Fire detection and Smoke detection modules .....	58
5.14	Fluid level detection .....	60
5.14.1	Functionality of the Fluid level detection module .....	60
5.14.2	Camera requirements for the Fluid level detection module .....	61
5.14.3	Configuring the Fluid level detection module.....	61

5.15	Neurotracker .....	63
5.15.1	Functionality of the Neurotracker module.....	63
5.15.2	Camera requirements for the Neurotracker module .....	64
5.15.3	Configuring the Neurotracker module.....	64
5.15.3.1	Configuring the neurotracker track counter .....	66
5.16	Person location tracker .....	68
5.16.1	Functionality of the Person location tracker module.....	68
5.16.2	Camera requirements for the Person location tracker module .....	68
5.16.3	Configuring the Person location tracker module.....	69
5.16.3.1	Configuring the pose detection.....	71
<b>6</b>	<b>Operating with detection modules .....</b>	<b>74</b>
6.1	Operating the «Queue length detection» module.....	74
6.1.1	Obtaining traffic information in the area of interest.....	74
6.1.2	Generating a report on the traffic in the area of interest.....	74
6.1.3	Visualization of operating the Queue length detection .....	74
6.2	Operating the «People counter detection» module.....	75
6.2.1	Obtaining information on number of visitors.....	75
6.2.2	Generating a visitor report .....	75
6.2.3	Visualization of operating the People counter detection .....	76
6.3	Operating the «Stopped vehicle detection» module .....	76
6.4	Operating the «Glow detection» module.....	77
6.5	Operating the «Heat map detection».....	77
6.5.1	Generating a report on the basis of Heat map detection .....	77
6.6	Operating the «Detection of moving against crowd flow».....	77
6.7	Operating the «Sweethearting detection» module.....	78
6.7.1	Generating sweethearting reports.....	78
6.8	Operating the «Barcode detection» module .....	78
6.9	Operating the «Train detection» module.....	79
6.10	Operating the "Fire detection" and "Smoke detection" modules .....	79
6.11	Operating the «Fluid level detection» module .....	80
6.12	Operating the «Neurotracker» module .....	81
6.13	Operating the Person location tracker module.....	81
<b>7</b>	<b>Appendix 1. Debug window .....</b>	<b>82</b>
7.1	General information .....	82

7.2	Start the debug window .....	82
7.3	Interface of debug window.....	83
<b>8</b>	<b>Appendix 2. Commands, requests and events of the Detector Pack software modules .</b>	<b>86</b>
8.1	Events of the software modules.....	86
8.2	Commands of the software modules.....	87

# 1 Introduction

## 1.1 General information

No part of this publication may be reproduced or transmitted in any way or in any form without the prior written consent of AxxonSoft.

The document contains information that is current at the time of publication. The document may be changed by AxxonSoft without prior notice to other parties.

## 1.2 The purpose of the document

The *Intellect software system – Intellect Detector pack: Operator's Manual* contains the information necessary to install and operate the additional software modules that are part of the *Intellect* software system detector pack.

The structure of this document allows the user to skim the information contained on the *Intellect Detector* pack and to select, depending on the level of training, topics of interest for a more detailed study. Chapters in the manual - or the informational or reference content – each have their own underlying structure.

The [Introduction](#) section is intended as a general introduction to this document.

The [Software and hardware requirements](#) section describes the requirements for computers and operating systems that use software modules included in the *Intellect Detector pac*.

The [Staffing requirements](#) section contains the requirements for personnel working with software modules included in the *Intellect Detector pack*.

The [Installing the Intellect detector pack](#) section contains the recommendations for users and administrators to install, repair and remove the *Intellect Detector pack*.

The [Configuring detection modules](#) section provides the information on functional characteristics, requirements for video cameras, and configuring the detection modules.

The [Operating with detection modules](#) section provides the information on operating the detection modules.

## 1.3 Purpose of the Intellect detector pack

The following *Intellect Detector pack* modules are intended for integration and use with *Intellect*.

1. **Queue length detection.**
2. **People counter detection.**
3. **Stopped vehicle detection.**
4. **Glow detection.**
5. **Heat map detection.**
6. **Detection of moving against crowd flow.**
7. **Barcode detection.**
8. **Train detection.**
9. **Smoke detection.**
10. **Fire detection.**
11. **Sweethearting detection.**
12. **Fluid level detection.**
13. **Neurotracker.**
14. **Person location tracker.**

The functional characteristics of these detection modules are provided in the corresponding sections (see [Configuring detection modules](#)).

## 2 Software and hardware requirements

The requirements for the base computers and operating system for the modules included in the *Intellect Detector pack* correspond to the same requirements for the *Intellect* (see [Intellect Administrator's Guide](#)).

Each software module automatically determines which computing resources it can use — a central processing unit (CPU) or a graphics processing unit (GPU).

For the modules that use neuroanalysis (see [General information on Neural analytics](#)), the operation device can be set manually.

Requirements for the hardware platform when the neuroanalysis is used:

1. The neural network analytics supports the following devices: CPU, GPU (NVidia discrete GPUs, integrated Intel GPUs), VPU (Intel NCS, Intel HDDL).

**Note.**

In order to connect the [Intel NCS](#), plug in the device to the USB port and make sure that Windows shows it with one of the following names: Movidius, Myriad X, VSC Loopback Device.

[Intel NCS](#) can be utilized with any computer compliant to the *Intellect* hardware requirements (see [Requirements for base PCs](#)).

**Attention!**

It is not recommended to use more than one [Intel NCS](#) device on the same Server.

**Attention!**

The neural network analytics functions are not available for AArch64 (ARM) architecture processors.

2. If CPU or Intel GPU is used for the neural network analytics operation, keep in mind that the following processors are supported: Intel Core generation 6 and later, Intel Xeon and Intel Pentium® processors N4200/5, N3350/5 or N3450/5 with Intel® HD Graphics <https://software.intel.com/en-us/opencv-toolkit/hardware>.
3. NVIDIA GeForce 1050 Ti GPU or newer. GPU requirements:
  - a. at least 2 GB of memory;
  - b. Compute Capability 3.0 or higher.

**Note.**

Please check the GPU's Compute Capability version on the [manufacturer's web site](#).

**Attention!**

When using NVidia graphics cards, it is recommended to install the latest driver from the [NVidia official web site](#).

A single neural network consumes 500 MB of video memory. Each neural network is distributed between several object trackers within the GPU. For example: to apply the neural fire detector and the neural smoke detector to any number of channels, you should use a 1 GB graphics card or higher. You can use multiple video cards in your system.

**Attention!**

Make sure that video image requirements are met in order for each detection module to operate correctly.

The requirements are specific for each detection module and are given in the corresponding sections (see [Configuring detection modules](#)).

## 2.1 Platforms for the detection software modules operation

The *Intellect Detector pack* is comprised of the independent software modules that operate on the following platforms:

Module	x32	x64
Heat map detection	✓	✓
Detection of moving against crowd flow	✓	✓
Queue length detection	✓	✓
Smoke detection	✗	✓
Fire detection	✗	✓
Stopped vehicle detection	✓	✓
People counter detection	✓	✓
Train detection	✓	✓
Glow detection	✓	✓
Sweethearting detection	✗	✓
Barcode detection	✓	✗
Fluid level detection	✗	✓
Neurotracker	✗	✓
Person location tracker	✗	✓

**Attention!**

Simultaneous operation of 32-bits and 64-bits modules on one PC is impossible.

**Note.**

When VMDA tracker process is separated to the single process, it operates correctly both in 32-bit and 64-bit platform. VMDA tracker configuration is described in the *Intellect software. Administrator's Guide*. Separating the VMDA tracker to the single process is performed with the VMDAEXT registry key which is described in the *Registry Keys Reference Guide*. The most recent versions of these documents are available in the [AxxonSoft documentation repository](#).

The basic version of *Intellect* is a software platform for the installation of these modules.

### 3 Staffing requirements

For the use of the *Intellect*-based detector, there are the following roles:

1. administrator;
2. operator.

In particular cases a person can perform the functions of both the administrator and operator. The main duties of the administrator are:

1. upgrading, configuring and monitoring the performance of the system hardware;
2. installing, upgrading, configuring and monitoring the performance of the system and basic software;
3. installing, configuring and monitoring the application software.

The administrator must have a high level of qualifications and practical experience in the implementation of the installation, configuration and administration of software and hardware used in the software package. The structure of the system provides the ability to control all functionality available by a single administrator, and also allows for the sharing of the administrative responsibility among multiple operators. The main duties of the operator are as follows:

1. work with the system's graphical user interface;
2. optimization of the PC for the tasks needed using the functionality provided in the system.

The system operator should have experience working with PC's based on Microsoft Windows operating systems at the level of a skilled user, and easily carry out basic operations.

## 4 Installing the Intellect Detector pack

### 4.1 General information on installing the Intellect Detector pack

The installation of the *Intellect detector pack* takes place in the following order:

1. Install *Intellect* (see the [Intellect system Administrator's Manual](#)).
2. Install the *Intellect Detector pack* (see the chapter on [Installation](#)).

### 4.2 Installing the Intellect Detector pack

#### 4.2.1 Description of the Intellect Detector pack installation files

The *Intellect Detector pack* installation files on CD-ROM.



The installation files contain the installation program and the necessary software components to install the *Intellect Detector pack* on the computer.

Only an administrator can install the *Intellect Detector pack*.

#### 4.2.2 Installation

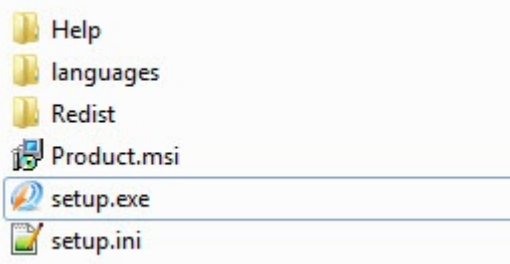
The *Intellect Detector Pack* software is installed as a part of the *Intellect* software. Information about compatibility of the *Intellect* software versions and *Intellect Detector Pack* is available at [General information about product releases and versions compatibility](#).

##### **Important!**

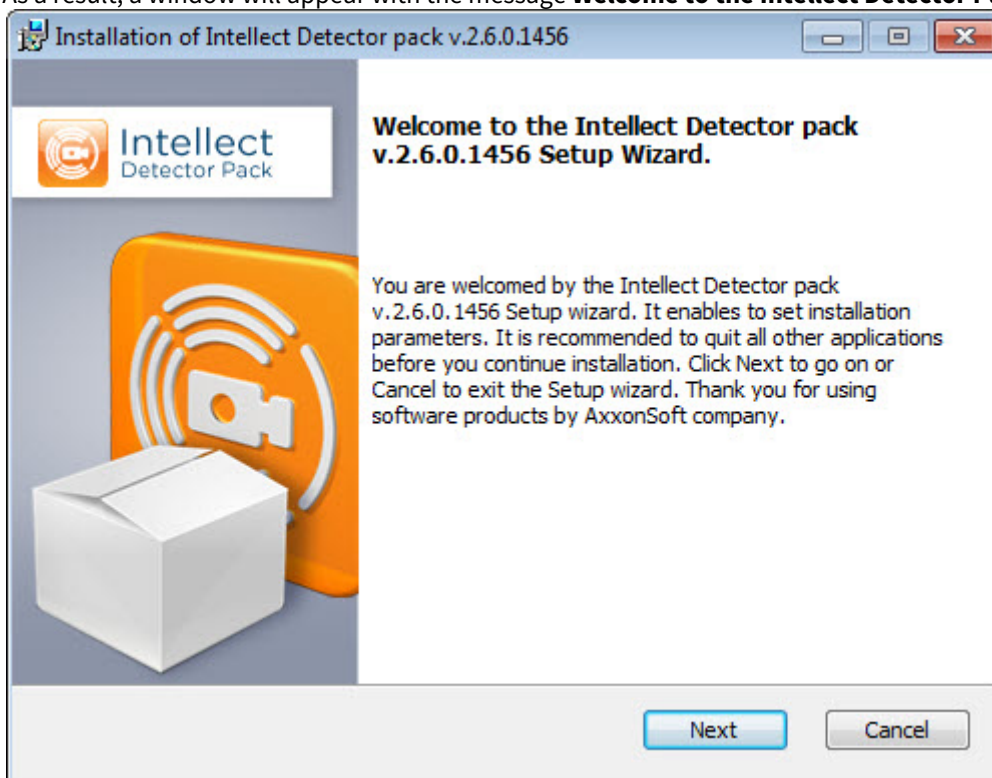
The *Intellect Detector Pack* software should be installed on both **Server/Remote administrator workplace** and **Client**. For details, see [Intellect. Administrator's Guide](#).

To install the *Intellect Detector pack*, the following steps must be performed:

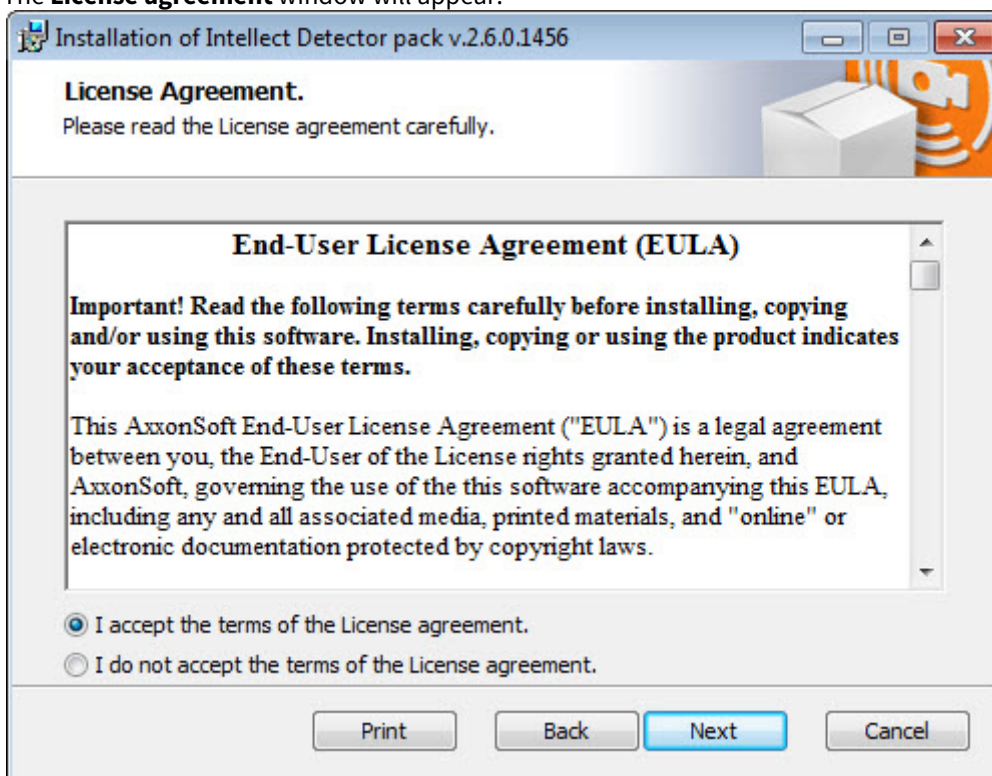
1. Insert the CD-ROM with the *Intellect Detector pack* installation files into the CD/DVD drive. A window will open showing the contents of the disc.



- Run **Setup.exe**, which will start the *Intellect Detector pack* installation.  
As a result, a window will appear with the message **Welcome to the Intellect Detector Pack v.2.0.0 Setup Wizard**.



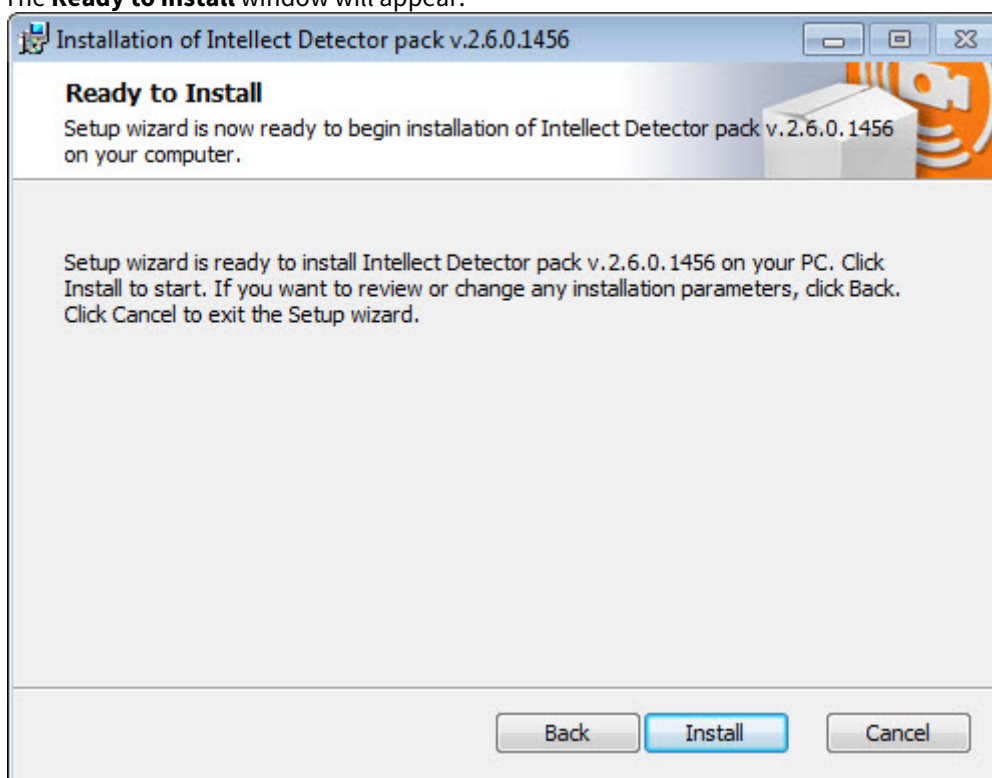
- Click **Next**.  
The **License agreement** window will appear.



- After reading the license agreement, agree with the terms of the agreement by clicking on the check box stating **I accept the terms of the License agreement**, otherwise the installation of the software system will be discontinued.

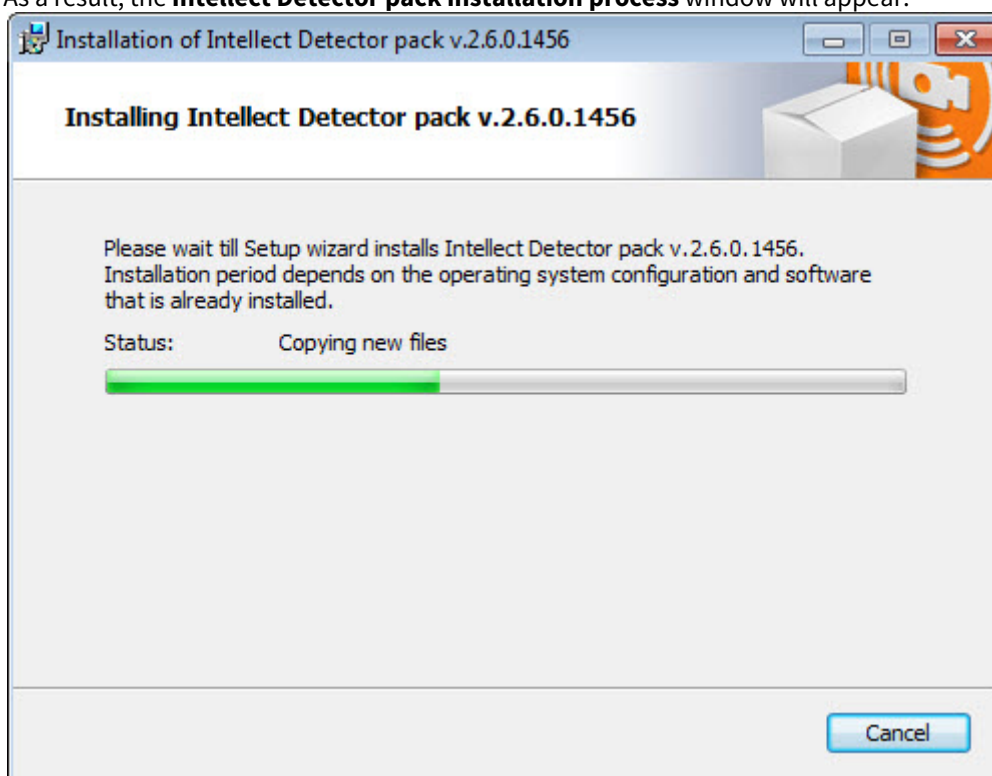
5. Click **Next**.

The **Ready to install** window will appear.

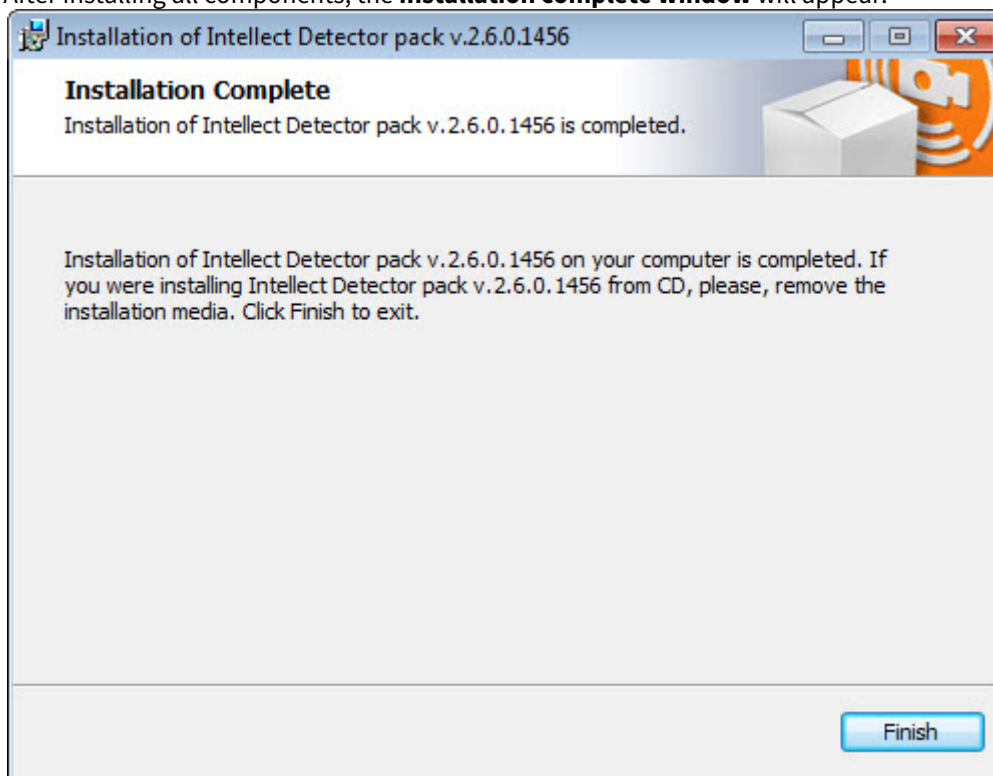


6. Click **Install**.

As a result, the **Intellect Detector pack installation process** window will appear.



After installing all components, the **Installation complete window** will appear.



7. Click **Finish**.

The *Intellect Detector pack* installation is complete.

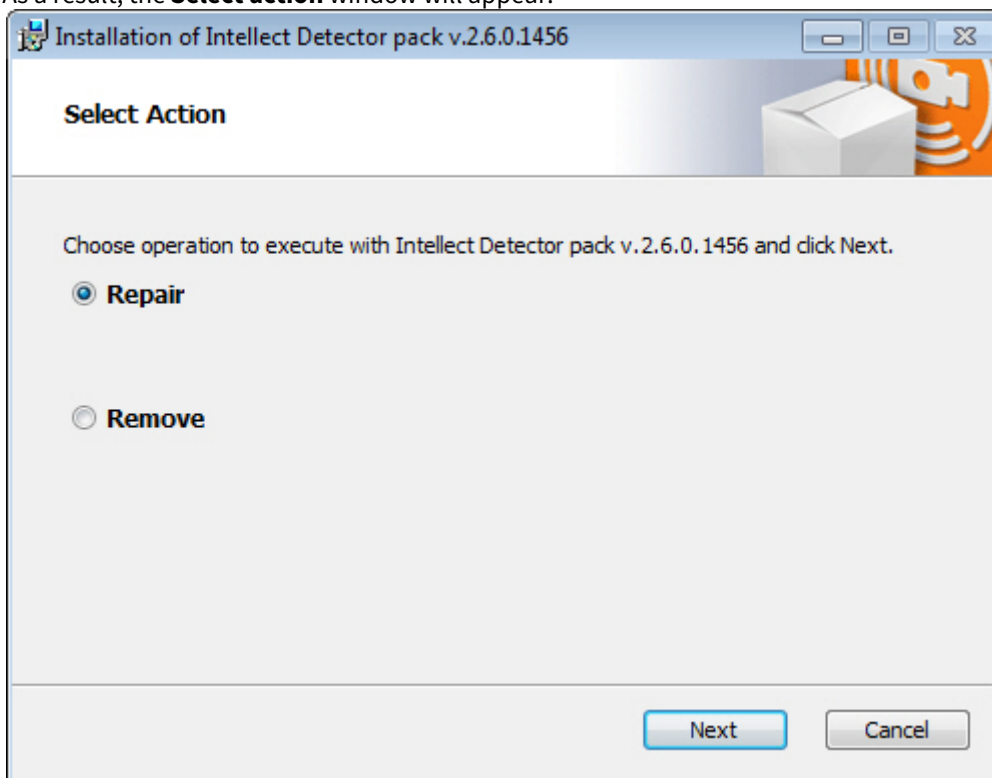
### 4.2.3 Repair

To repair the *Intellect Detector pack*, the following steps must be carried out:

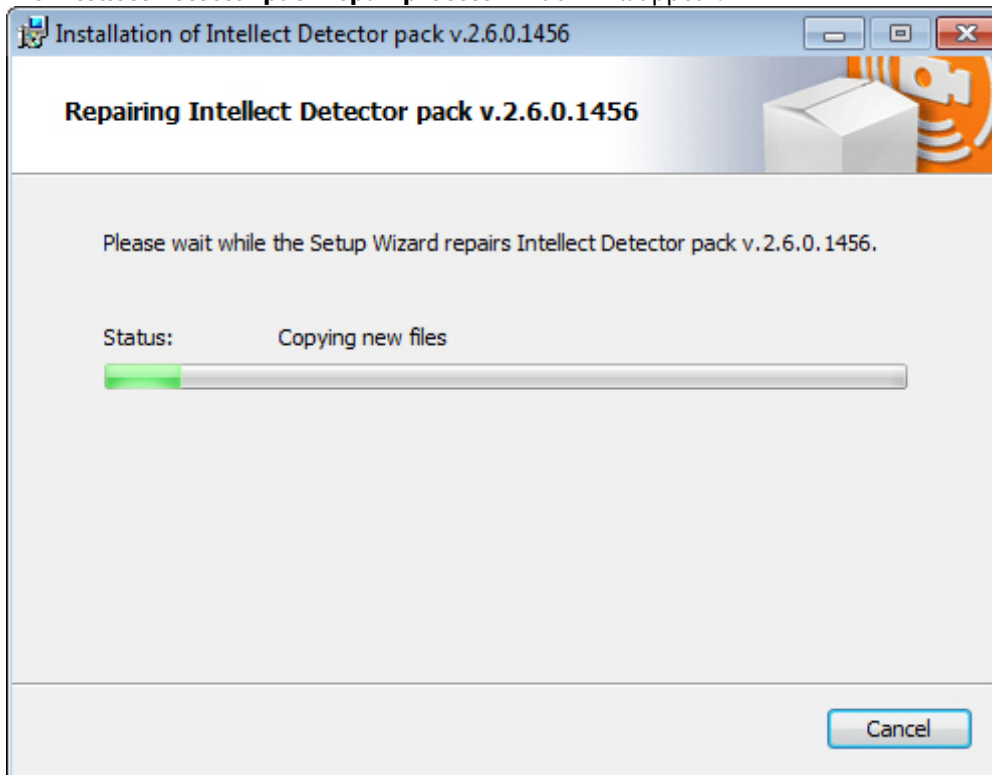
1. Insert the CD-ROM with the *Intellect Detector pack* installation files into the CD/DVD drive. A window will open showing the contents of the disc.



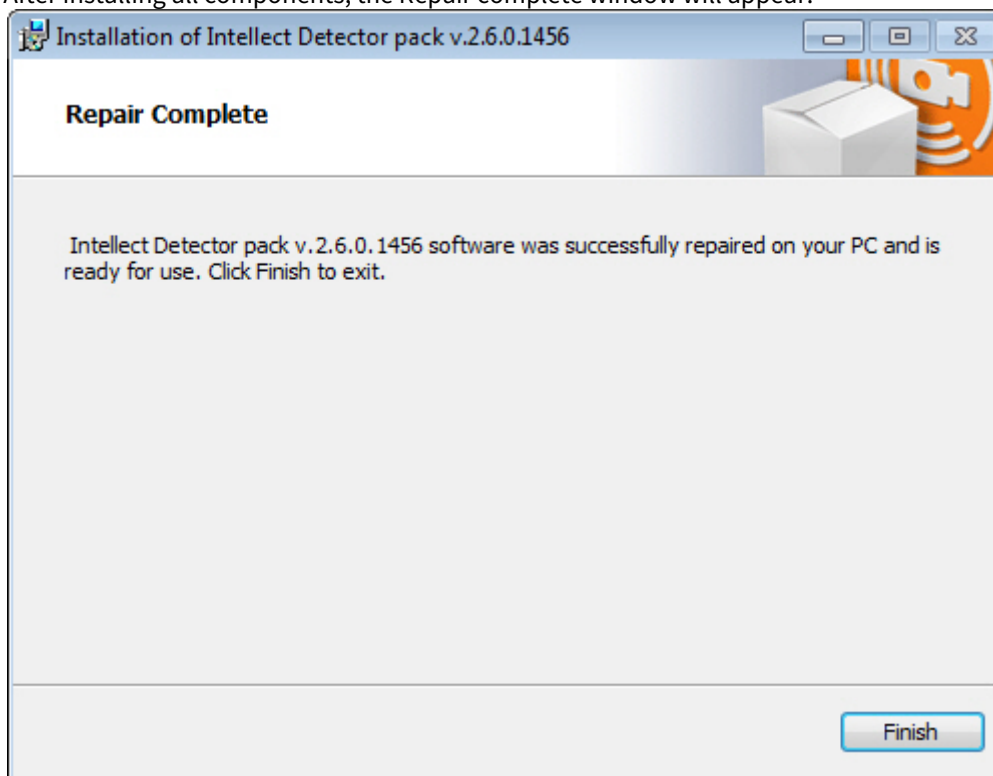
2. Run **Setup.exe**, which will start the *Intellect detector pack* installation. As a result, the **Select action** window will appear.



3. Select **Repair**.
  4. Click **Next**.
- The **Intellect Detector pack repair process** window will appear.



After installing all components, the Repair complete window will appear.



5. Click **Finish**.

The *Intellect Detector* pack repair is complete.

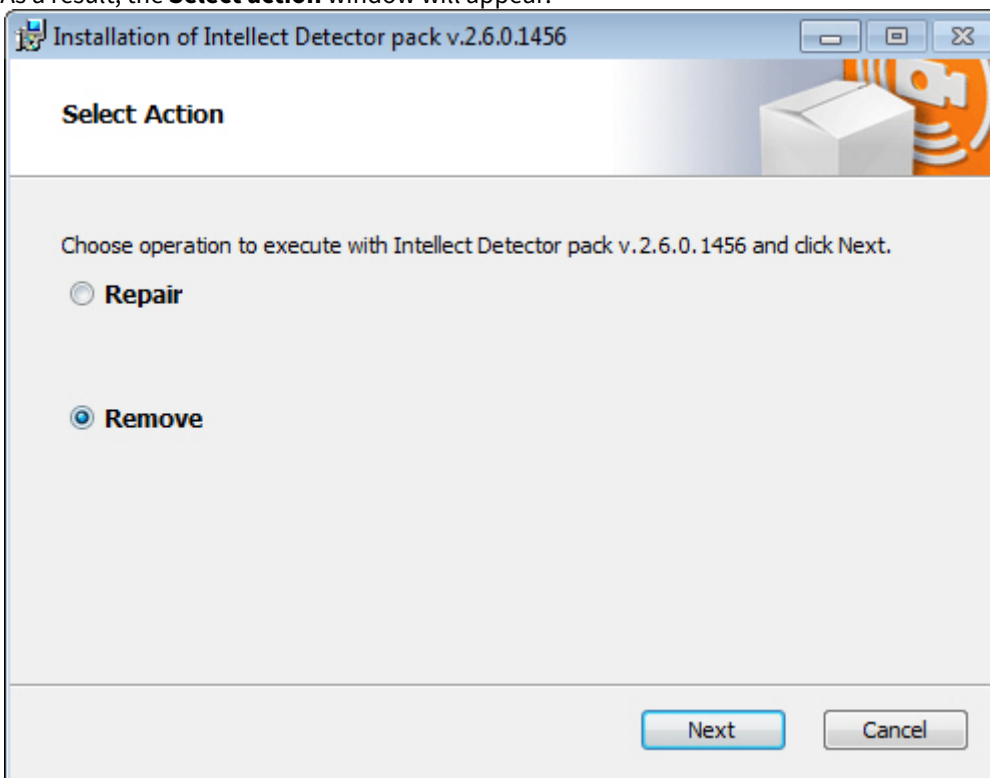
## 4.2.4 Removal

To remove the *Intellect Detector* pack, the following steps must be carried out:

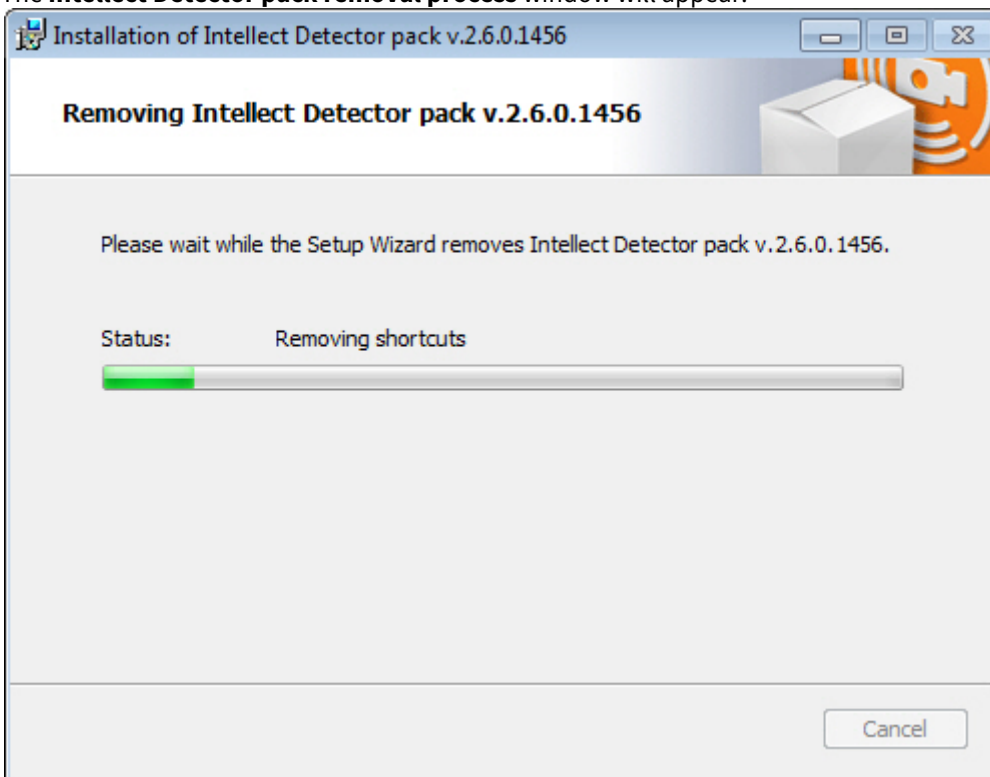
1. Insert the CD-ROM with the *Intellect Detector* pack installation files into the CD/DVD drive. A window will open showing the contents of the disc.



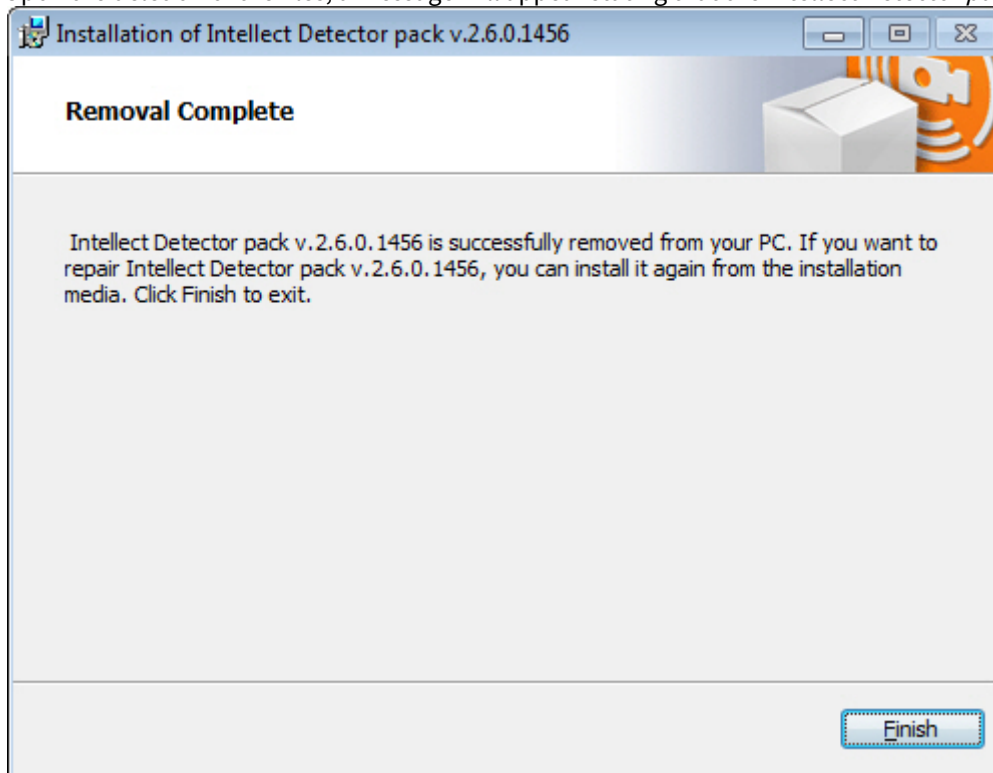
- Run **Setup.exe**, which will start the *Intellect Detector pack* installation. As a result, the **Select action** window will appear.



- Click **Remove**.
  - Click **Next**.
- The **Intellect Detector pack removal process** window will appear.



Upon the deletion of the files, a message will appear stating that the *Intellect Detector pack* was removed.



5. Click **Finish**.

The *Intellect Detector pack* removal is complete.

#### 4.2.5 Installing the Intellect Detector Pack in quiet (silent) mode

It is possible to install *Intellect Detector Pack* in a quiet (silent) mode. For this select one of the following options:

1. Specify in the **setup.ini** file in the installation directory (see [Installation](#)):

```
[Startup]
CmdLine=/quiet

[Info]
Name=DetectorPack
Version=1.00.000
DiskSpace=8000 ;DiskSpace requirement in KB

[Startup]
CmdLine=/quiet
Product=Intellect Detector Pack
PackageName=Product.msi
LogOptions =
QuietCmdLine=/quiet

[Install1]
DisplayName=MS Visual Studio 2015 Redistributable x64
CmdLine=/q
Product=Redist\VC2015_Redistributable_Package\x64\vcredist_x64.exe
Required=1
QuietCmdLine=/q
Platform=x64
GUID={A1C31BA5-5438-3A07-9EEE-A5FB2D0FDE36}
```

When **Setup.exe** is run, *Intellect Detector Pack* is installed in the quiet (silent) mode.

2. In the *Intellect Detector Pack* installation directory specify the following command in the Windows command line:

```
setup.exe /quiet
```

When the command is executed, the *Intellect Detector Pack* installation in the silent mode starts.

## 5 Configuring detection modules

### 5.1 General information on Neural analytics

The detector pack uses the AI analytics based on neural networks. The following AI features are available:

1. **Neurotracker** (see [Neurotracker](#)).

Neural Tracker detects only objects of a specified class. The Neural Tracker is more accurate than the regular one, and detects even static objects, but it requires more computing resources.

**Note**

The quality of work and the resource consumption of the neurotracker directly depend on the optimization of the neural network model used.

Before you start setting up the neural tracker, you should contact the [AxxonSoft](#) technical support and request the model files of the trained neural networks. Technical support specialists will request from you the data needed to prepare the models, and then will provide you with the **\*.ann** and **\*.txt** files for each neural network model. These files should be distributed to all Servers where it is planned to use the neurotracker.

2. **AI Smoke and Fire detection tools** (see [Fire detection and Smoke detection](#)).

A neural network detects fire and smoke in FoV.

**Note**

The quality of work and the resource consumption of the AI Smoke and Fire detection tools directly depend on the optimization of the neural network model used.

Before you start setting up the Smoke and Fire AI detection tools, you should contact the [AxxonSoft](#) technical support and request the model files of the trained neural networks. Technical support specialists will request from you the data needed to prepare the models, and then will provide you with the **\*.ann** files for each neural network model. These files should be distributed to all Servers where it is planned to use the AI Smoke and Fire detection tools.

3. **AI Person location tracker** (see [Person location tracker](#)).

AI-powered Posture Detection captures specific human poses that may represent a security threat.

Software and hardware requirements

### 5.2 Selecting a camera video stream for the detection modules

By default, the detection modules included in the *Detector Pack* subsystem use the Video Analytics camera stream. In order for the detection modules to use the Default camera stream, it is necessary to change the **VMDA.ignoreAnalyticStream** registry key value to **1** (see [Registry keys reference guide](#), for details about working with the registry, see [Working with Windows OS registry](#)).

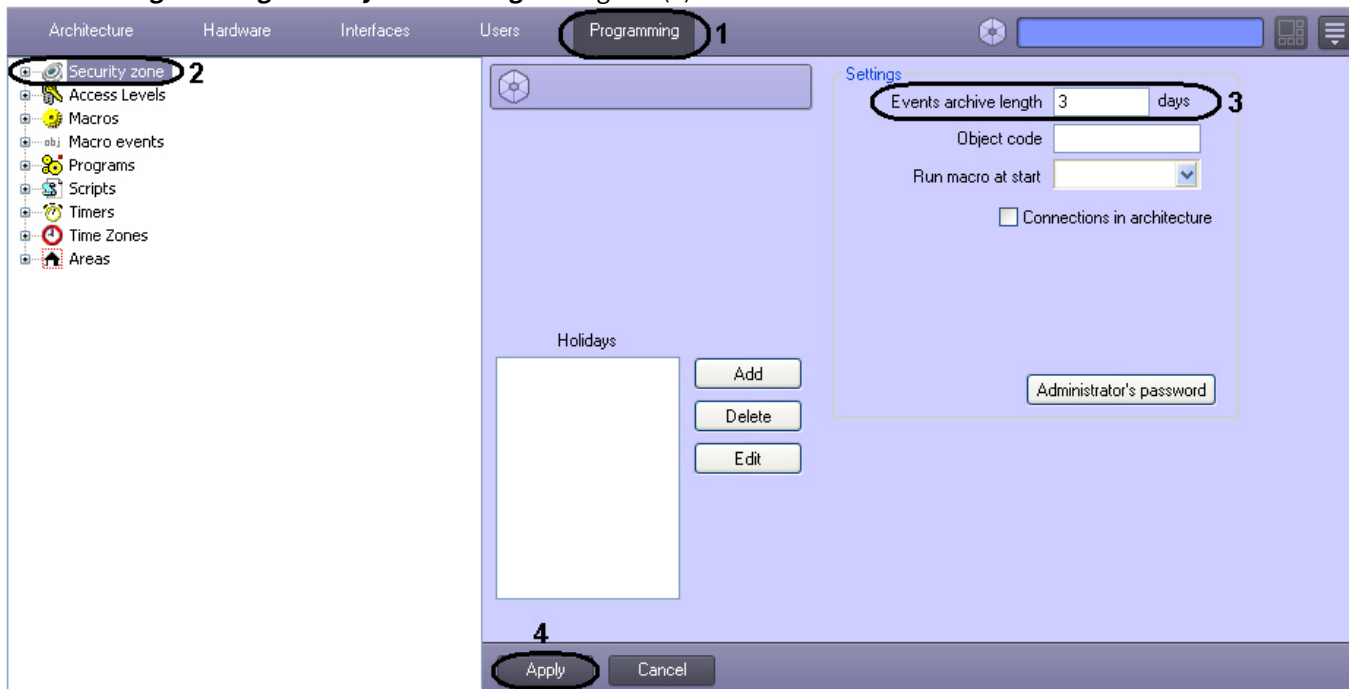
**Note**

If the camera is not used in the multi-thread mode, the first stream (the only one) will be used regardless of the key value.

## 5.3 Configuring the events archive length

Reports by results of detection working received with the help of *Report System* web-reports subsystem are creating on information from the event log database. On default the event archive is storing in the database for three days, but it is insufficient for proper report creating. To increase the events archive length, do the following:

1. Go to the **Programming** tab of **System settings** dialog box (1).



2. Go to the **Security zone** object's settings panel (2).
3. Enter the period of events storage in the database in the **Events archive length \_\_\_ days** field (3). For example, specify the archive storage period equal to 30 days.
4. Click the **Apply** button (4).

Configuring the events archive length is completed.

## 5.4 Queue length detection

### 5.4.1 Functionality of the Queue length detection module

The **Queue length detection** module is designed to carry out the following functions:

1. Count the number of people waiting in line within a certain time interval.
2. Record the number of people waiting in line in a database.
3. Plot the crowding in an observed area.
4. Generate an event when threshold queue length is exceeded and record it to the Event protocol database.

### 5.4.2 Camera requirements for the Queue length detection module

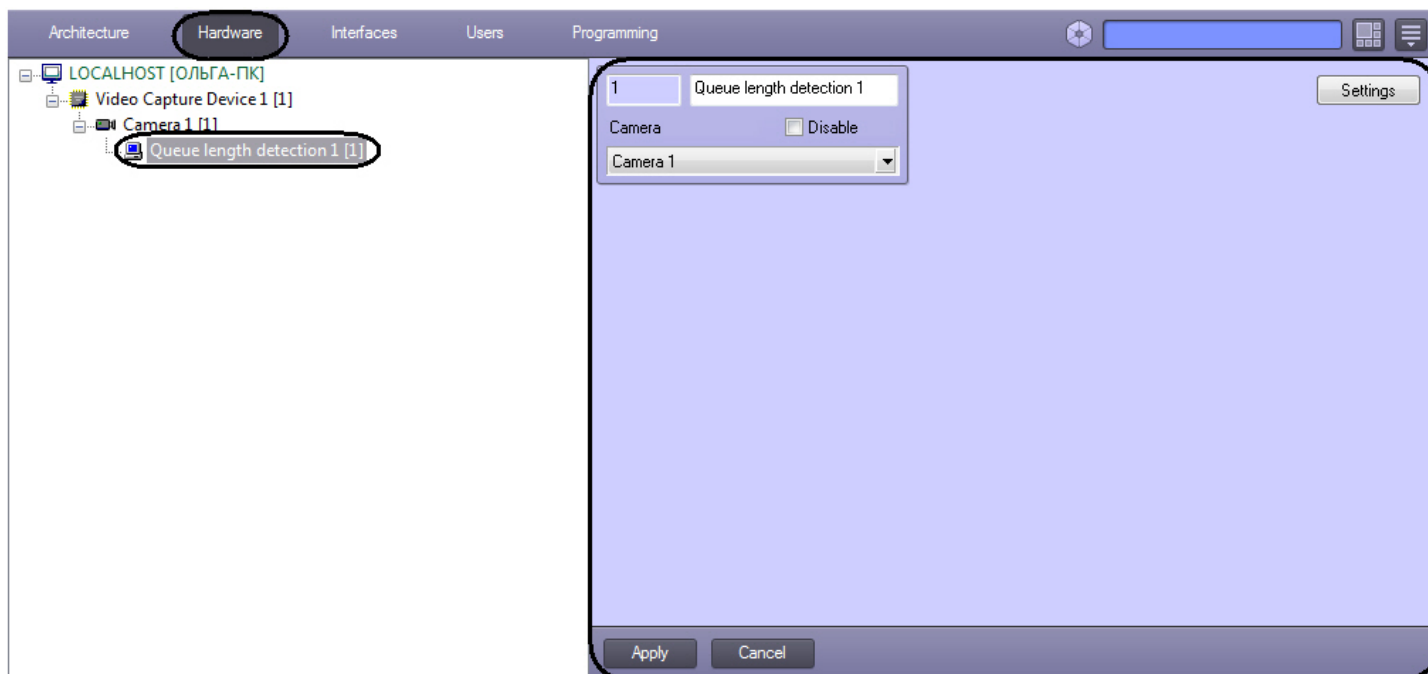
The requirements for the cameras that will work with the **Queue length detection** module are listed in the following table.

Camera	<ul style="list-style-type: none"> <li>• Resolution: 720x576 (CIF4), using of 360x288 (CIF1) is acceptable; oversize images are reduces until CIF4.</li> <li>• Fps: not less than 6</li> <li>• Color: analytics works with grey and color images.</li> <li>• Camera must be rigidly fixed.</li> </ul>
--------	---

<p>Lighting</p>	<ul style="list-style-type: none"> <li>• The best working of detection is archived at medium lighting. In conditions of insufficient (night) or excessive (exposure) lighting, the quality of analytics can be reduced.</li> <li>• Sharp changes of lighting can lead to short-time invalid analytics working.</li> </ul>
<p>Scene and camera angle</p>	<ul style="list-style-type: none"> <li>• The best position – camera "looks" to the scene vertically down. The better this requirement, the carefully the received estimation.</li> <li>• Sizes of camera field of view: 3x3m is minimal (6x6 people), 4x4m is optimal (8x8 people), 8x8m is maximal (16x16 people).</li> <li>• Background is static and is not changed sharply.</li> <li>• Analytics can work inappropriately on specular surfaces and in case of sharp shadows from moved objects.</li> <li>• Analytics can work inappropriately in case of in the camera field of view there are periodic movements of background objects (trees, working TV, etc.)</li> </ul>
<p>Objects image</p>	<ul style="list-style-type: none"> <li>• Image quality: the image is to be clear, without visible defects from reducing procedure.</li> <li>• Permissible size of a person (the area of a rectangular track around a person) as a percentage of the frame area: from 0.25% to 10%</li> </ul>

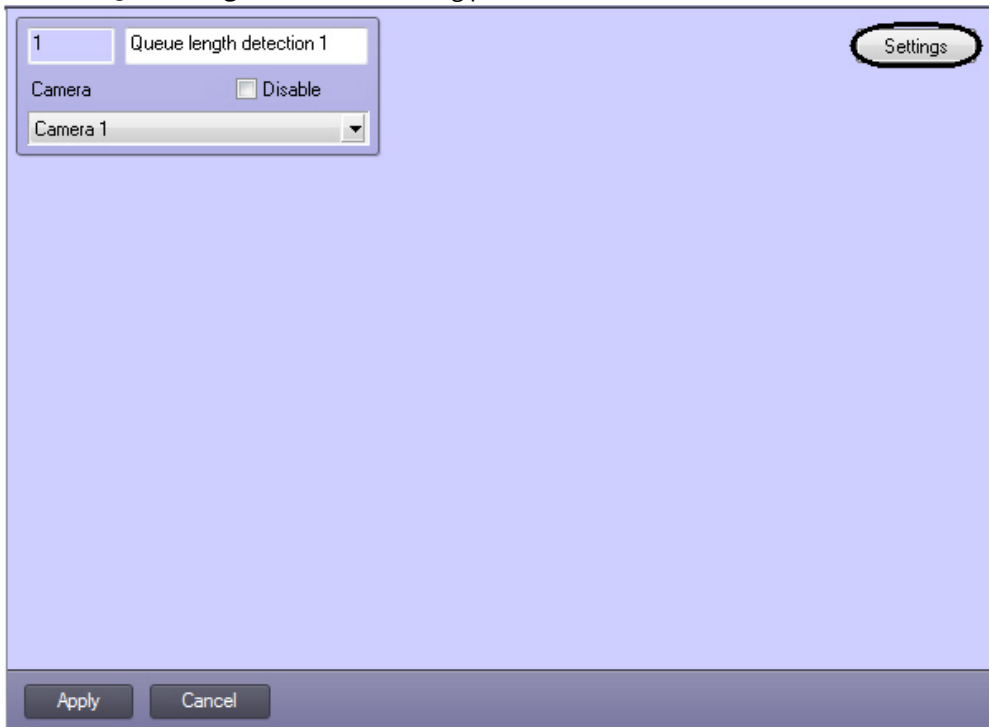
### 5.4.3 Configuring the Queue length detection module

The **Queue length detection** module can be configured on the settings panel of the **Queue length detection** object created under the **Camera** object in the **Hardware** tab of the **System settings** dialog box.

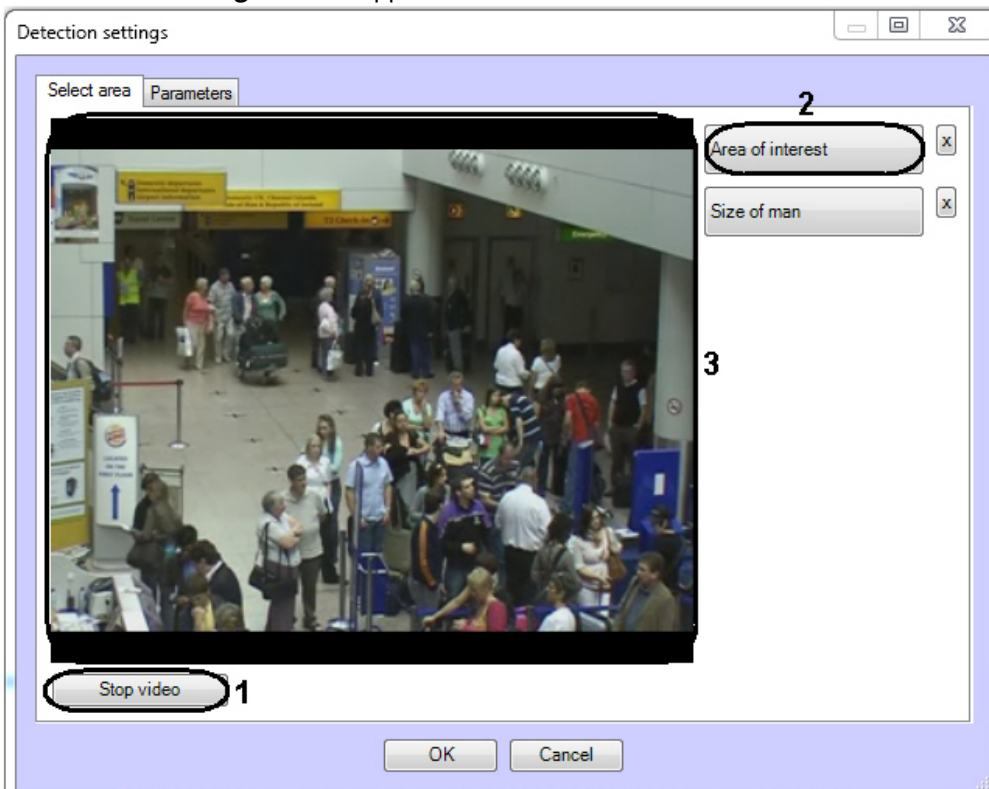


Configure the **Queue length detection** module as follows:

1. Go to the **Queue length detection** setting panel.

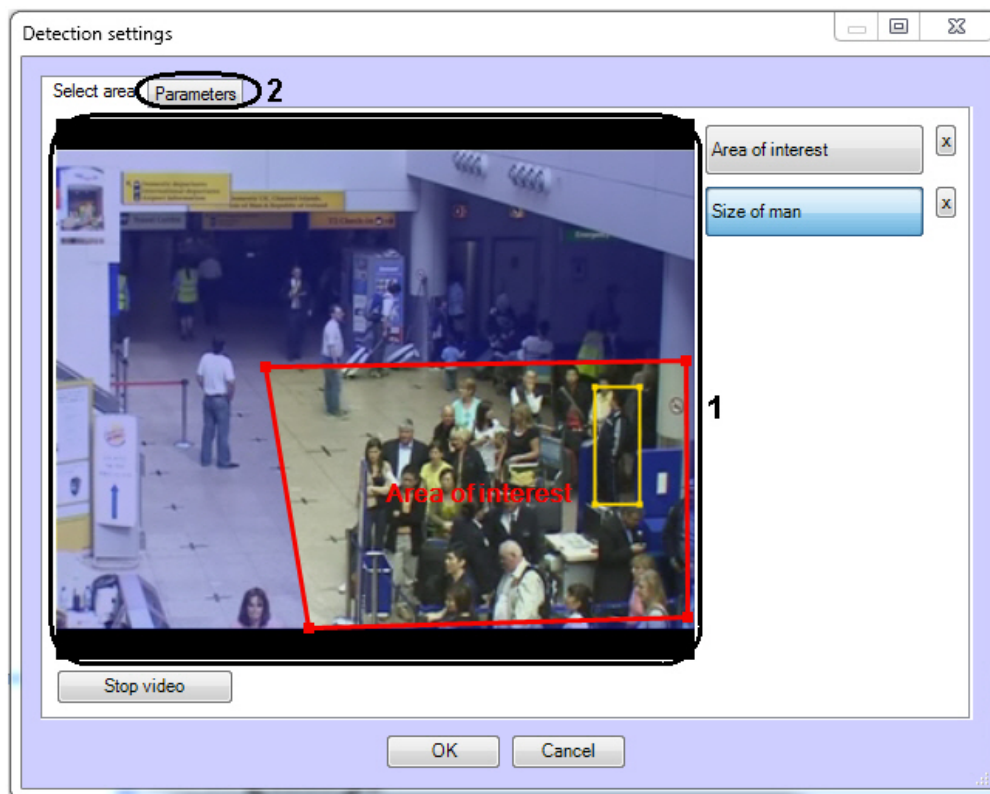


2. Click **Settings**.  
The **Detection settings** window appears.

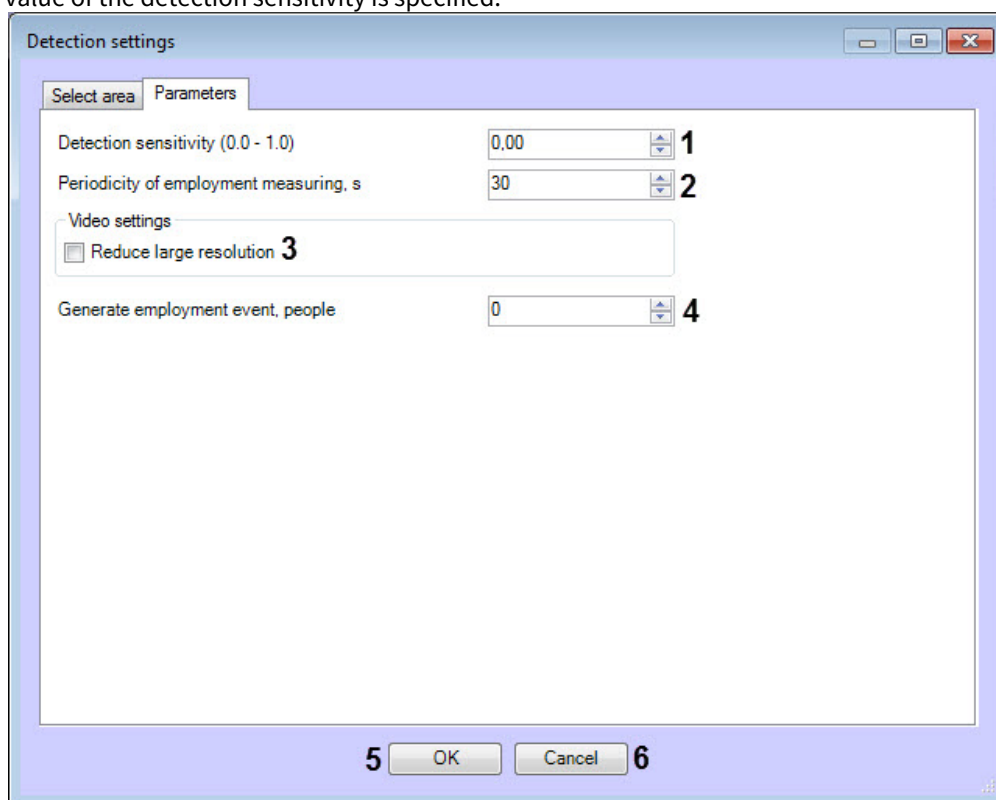


3. Specify the area of interest and the approximate size of a person in the video image:
  - a. Click **Stop video** to capture the video image (1).
  - b. Click **Area of interest** (2).
  - c. Using the left mouse button select the four points of the area in the captured video image (3) to be analyzed (1). Only one area can be added. If a second area is specified, then the first area will be deleted. Upon selection of the area the remaining part of the video image will be dimmed.





4. Set the module parameters:
  - a. Go to the **Parameters** tab (2).
  - b. Set the detection sensitivity field in a range from 0 to 1 with up to two decimal places (1). The higher sensitivity is, the less significant objects will be considered as a queue. i.e. the algorithm will react to more inconsiderable motion. So, only considerable change of the scene will be analyzed by the queue length detection if the minimum value of the detection sensitivity is specified.



- c. In the **Periodicity of employment measuring, s** field, specify a time period in seconds for counting the number of people in the observed area (2). The minimum value of this parameter is an interval between frames of analyzed video stream, but not less than 1 second and not more than 3600 seconds. The value of the parameter depends on the scene characteristics. For example, if a queue in the field of video camera view is not changing for a long time, than the parameter value can be reasonably large. It is not recommended to specify value of this parameter more than 3-5 minutes for correct working of algorithm on the average scene.

**Note**

The more frequently periodicity of employment is measured, the more frames are analyzed by algorithm and the higher the system capacity is.

- d. Set the **Reduce large resolution** checkbox to create and process the new frame consisting of even lines of the initial frame (3).
- e. In the **Generate employment event, people** field specify the value of the Queue length detection counter (number of people). If this value is exceeded, then the *Queue length* (OCCUPANCY\_EVENT) event is generated by the **Queue length detection** object (4). The event have the following parameters: current counter value (occupancy<>) and the threshold value (raise\_count<>). The event can be used in scripts and programs (see *Programming Guide (JScript)* and *Programming Guide* for more info on scripts and programs; the most relevant versions of these documents can be found in [AxxonSoft documentation repository](#)).

**Note**

All events from the **Queue length detection** object can be found in the **Event log**. See [Obtaining traffic information in the area of interest](#).

5. Click **OK** to save changes and return to the settings panel of the **Queue length detection** object (5).

**Note**

To return to the settings panel of the **Queue length detection** object without saving changes, click **Cancel** (6).

6. Click **Apply** on the settings panel of the **Queue length detection** object.

Configuring the **Queue length detection** module is completed.

## 5.5 People counter detection

### 5.5.1 Functionality of the People counter detection module

The **People counter detection** module is designed to carry out the following functions:

1. Count visitors in an observed area.
2. Record events about visitor entries into the observed area to a database.
3. Record events about visitor exits from the observed area to a database.
4. Generate reports by the number of visitors in the observed area.

**Note**

The **People counter detection** module does not operate properly in the real time mode, i.e. it cannot be used to immediately count visitors. As the operation algorithm of the detection tool is static, the maximum people count accuracy is reached at some periods of time (every 15 minutes of non-stop operation). Mainly people count accuracy depends on whether the requirements given in the [Camera requirements for the People counter detection module](#) are met.

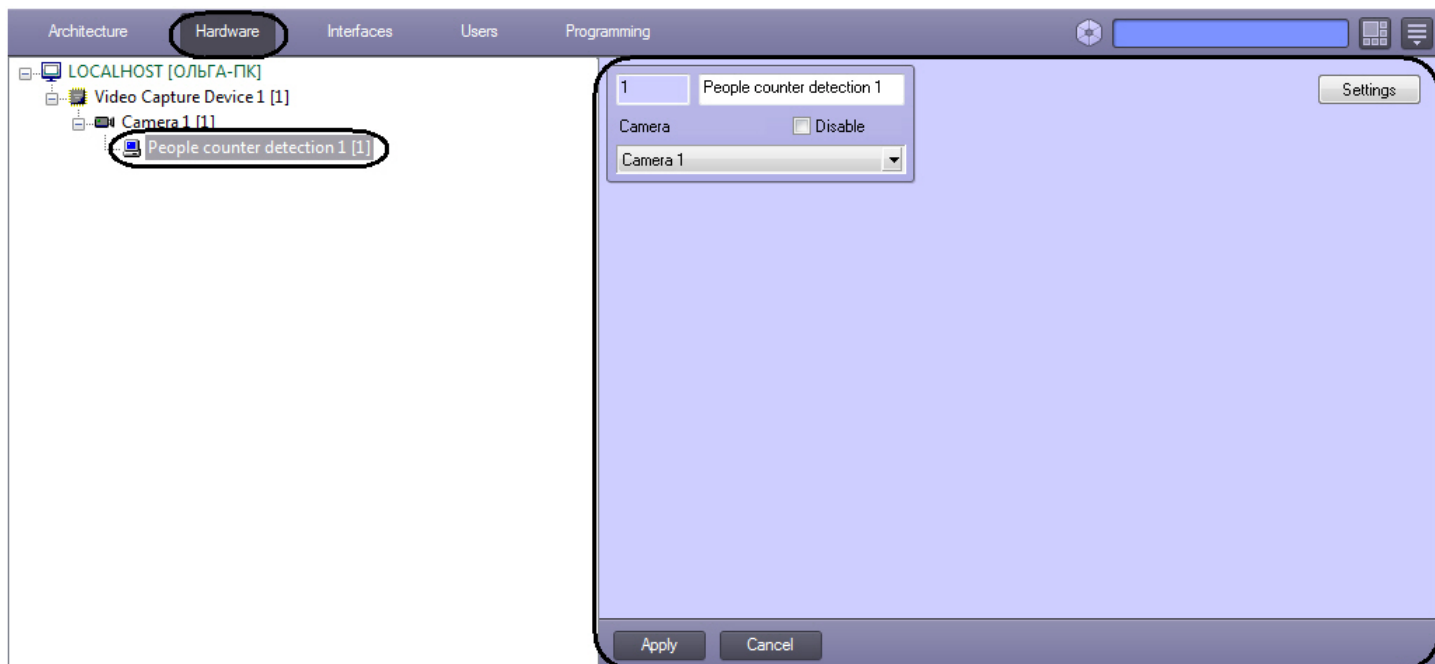
## 5.5.2 Camera requirements for the People counter detection module

The requirements for the cameras that will work with the **People counter detection** module are listed in the following table.

Camera	<ul style="list-style-type: none"> <li>Resolution: 720x576 (CIF4), using of 360x288 (CIF1) is acceptable. Resolution zoom-in over CIF4 is not improve the quality of recognizing procedure.</li> <li>Fps: 25.</li> <li>Color: only color camera can be in use.</li> <li>Camera must be rigidly fixed.</li> </ul>
Lighting	<ul style="list-style-type: none"> <li>The best working of detection is archived at medium lighting. In conditions of insufficient (night) or excessive (exposure) lighting, the quality of analytics can be reduced.</li> <li>Sharp changes of lighting can lead to short-time invalid analytics working.</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>The best position – camera "looks" to the scene vertically down. The better this requirement, the carefully the received estimation.</li> <li>Sizes of camera field of view: 2x2m is minimal, 4x4m is optimal (8x8 people).</li> <li>Background is static and is not changed sharply.</li> <li>In the recognized are there no moving objects except of people.</li> <li>Analytics can work inappropriately on specular surfaces and in case of sharp shadows from moved objects.</li> <li>Analytics can work inappropriately in case of in the camera field of view there are periodic movements of background objects (trees, working TV, etc.)</li> <li>People occulting by static objects is to be minimal (by columns, trees etc.).</li> </ul>
Objects image	<ul style="list-style-type: none"> <li>Image quality: the image is to be clear, without visible defects from reducing procedure.</li> <li>Permissible size of a person (the area of a rectangular track around a person) as a percentage of the frame area: from 10% to 60%.</li> </ul>
Other:	<ul style="list-style-type: none"> <li>People shouldn't move by continuous flow, by groups by several people are counted properly.</li> </ul>

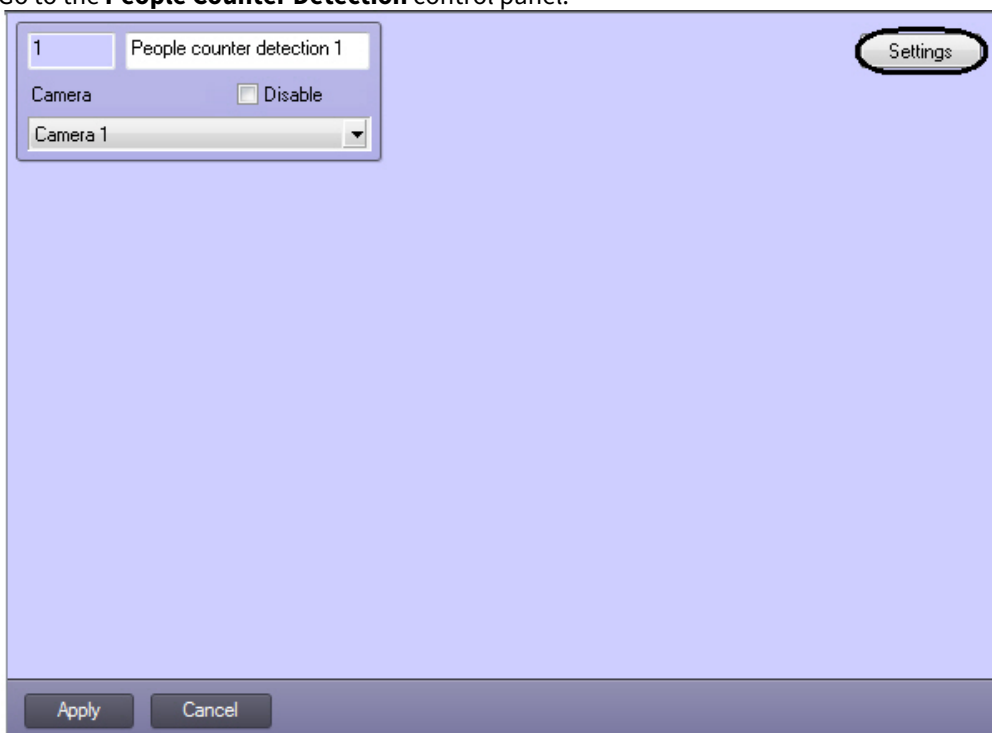
## 5.5.3 Configuring the People counter detection module

The **People counter detection** module can be configured using the **System settings** menu, under the **Hardware** tab, on the **People counter detection** control panel, using the **Camera** settings.

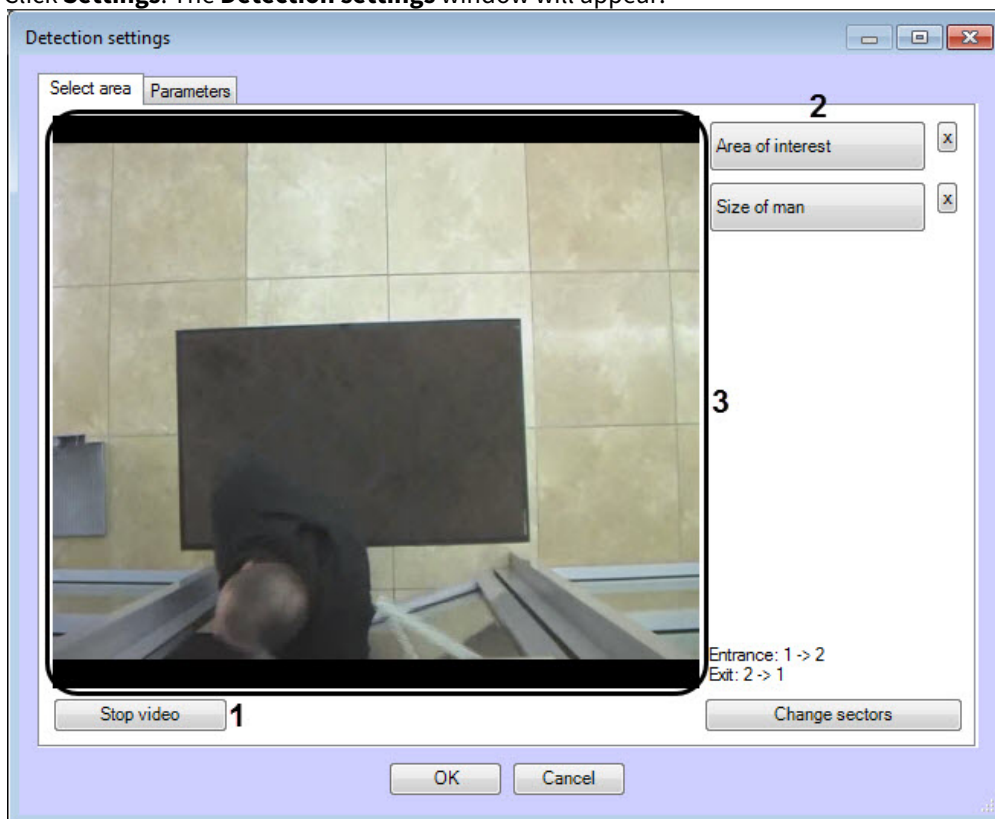


The **People counter detection** module is set up as follows:

1. Go to the **People Counter Detection** control panel.




2. Click **Settings**. The **Detection settings** window will appear.



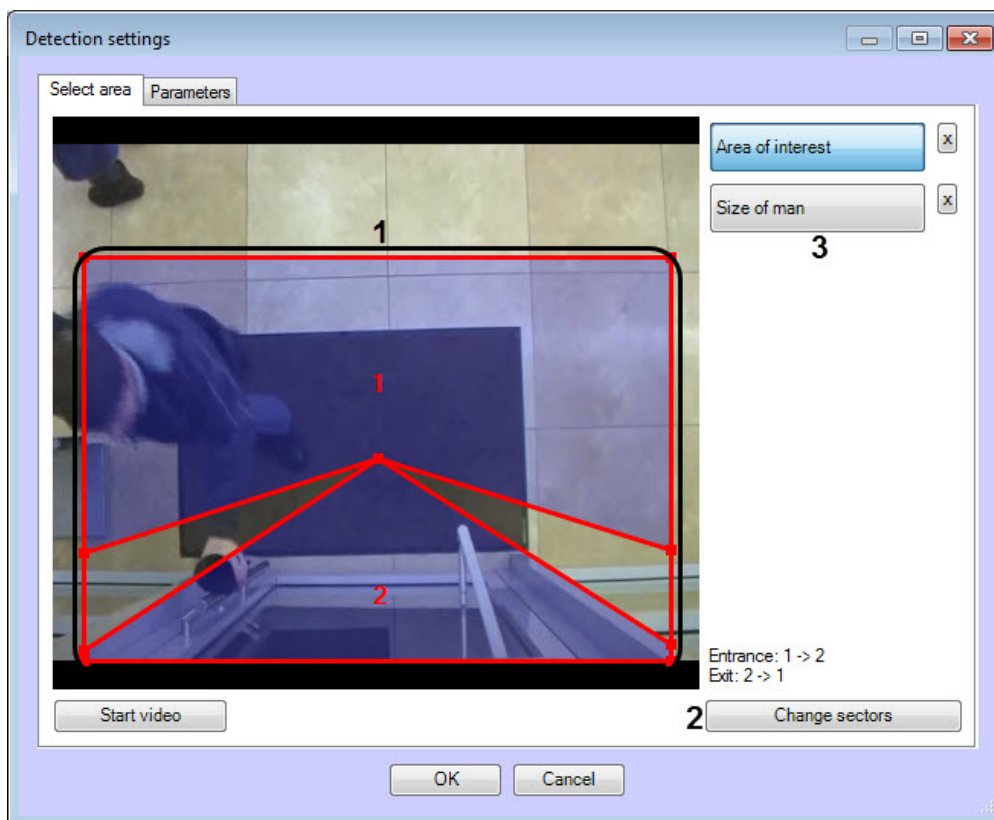
3. Specify the area of interest and the approximate size of people in the video image:
- Click **Stop video** to capture the video image (1).
  - Click **Area of interest** (2).
  - Using the left mouse button select the four corners of the area on the captured video image (3) to be analyzed. Only one area may be so designated. If a second area is specified, then the first area will be deleted.

**Note.**

To remove a selected area, click the  button next to the **Area of interest** button.

**Note.**

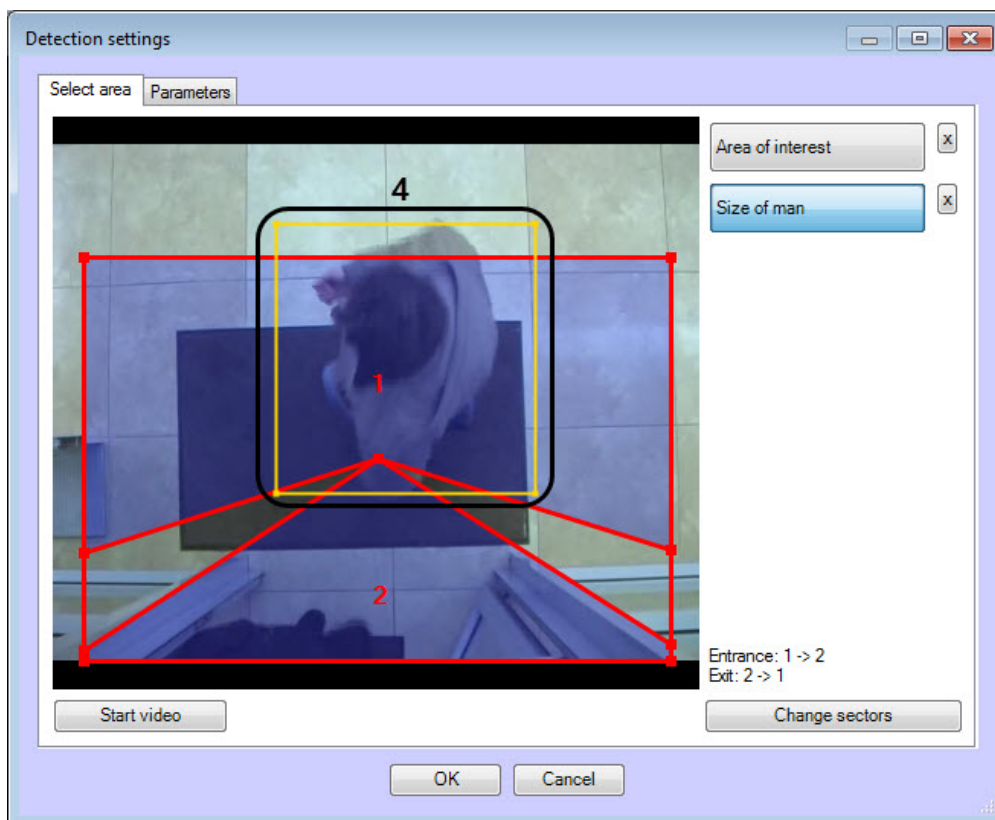
The area of interest is divided into two sections - 1 and 2. If an object moves from sector 1 to sector 2, it is logged as the entry of a visitor; if the visitor moves from sector 2 to sector 1, it is logged as an exit.



- d. Set the desired size, shape and position of the sectors in the area of interest by moving their boundaries (1).
  - e. If you want to swap sectors 1 and 2, click **Change sectors** (2).
4. Set the approximate person size as follows:
    - a. Click on **Person size** (3).
    - b. On the captured video image set the approximate person size. To do this, use the left mouse button to select a rectangular area (4).

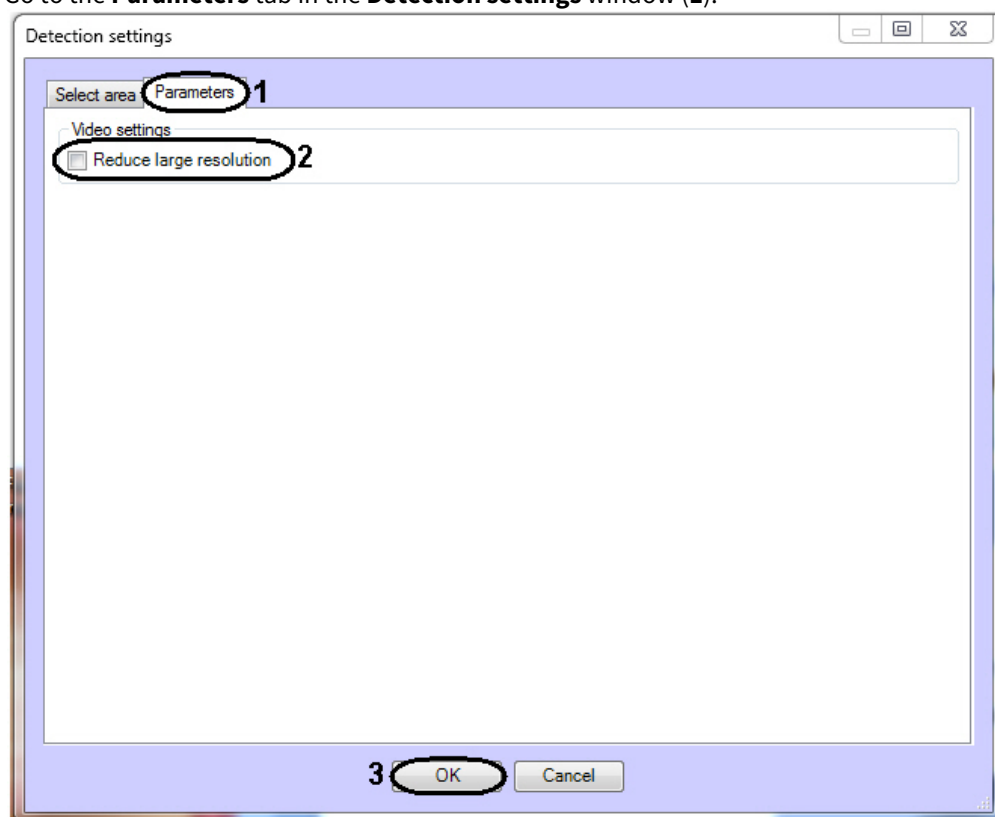
**Note.**

To remove the person size, click the  button next to the **Person size** button.



5. Setting the module parameters:

- a. Go to the **Parameters** tab in the **Detection settings** window (1).



- b. Set the **Reduce large resolution** checkbox to create and process the new frame consist of even lines of initial frame (2).

6. Click **OK** (3).

Configuring the *People counter detection* module is complete.

## 5.6 Stopped vehicle detection

### 5.6.1 Functionality of the Stopped vehicle detection module

The **Stopped vehicle detection** module is designed to carry out the following functions:

1. Recognizing cars stopped in the specified areas.
2. Recognizing jams in the specified areas.
3. Recording Jams and Stopped cars events to the database.
4. Recording events of jam elimination or start of movement of previously detected stopped car to the database.

### 5.6.2 Camera requirements for the Stopped vehicle detection module

The requirements for the cameras that will work with the **Stopped vehicle detection** module are listed in the following table:

Camera	<ul style="list-style-type: none"> <li>• Resolution should be at least 720x480 pixels</li> <li>• Fps: not less than 15, recommended fps is 25</li> <li>• Camera should be rigidly fixed to avoid shaking</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>• Recommended height of camera mounting: 6-7 m</li> <li>• Recommended angle of camera mounting: no more than 30 degrees relative to the horizontal line</li> <li>• It is recommended to mount camera above the middle of analyzed traffic area</li> <li>• If the camera is installed on the side of the traffic area, the efficiency of the detection algorithm decreases</li> </ul>
Object images	<ul style="list-style-type: none"> <li>• The vehicles and their elements, for example, a radiator grill, should be visually distinguishable on the video image</li> <li>• It is recommended to use the Zoom-In/Zoom-Out camera functions to improve the response reflection from vehicles on the scene, i.e. settings in which the vehicle elements are clearly distinguishable</li> </ul>
Lighting requirements	<ul style="list-style-type: none"> <li>• Objects should be visually separated from the background and from each other.</li> </ul>

### 5.6.3 Configuring the Stopped vehicle detection module

#### 5.6.3.1 Licensing the Stopped vehicle detection module

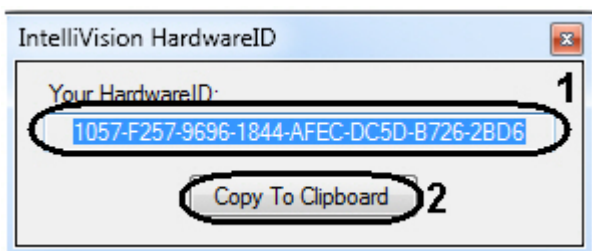
To license the «Stopped vehicle detection» module do the following:

1. Go to the *<Intellect installation folder>\Modules\IntelliVision* folder.
2. Run the *HardwareID.exe* utility.

#### **Attention!**

Start the *HardwareID.exe* utility from the name of computer Administrator.

Also the utility is to be started after the full loading of operating system, specifically after start of all needed services and applications.

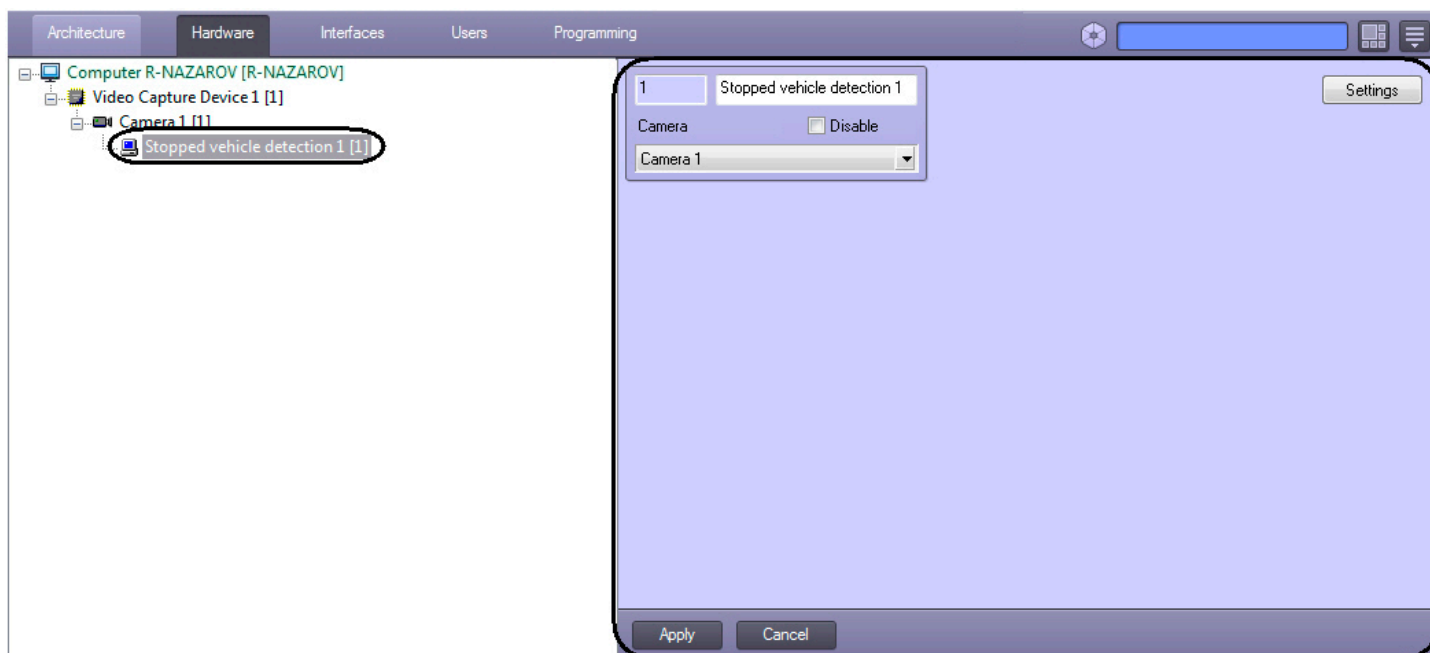


3. In the **Your HardwareID** field the code is displayed (1).
4. Click the **Copy To Clipboard** button to copy the code to the clipboard (2).
5. Send the code to the manager of the ITV company and specify the number of **Stopped vehicle detection** which are planned to be used.
6. Receive the regkey.dat file from the manager of the ITV company.
7. Put the received file to the <Intellect installation folder>\Modules folder.

Licensing the «Stopped vehicle detection» module is completed.

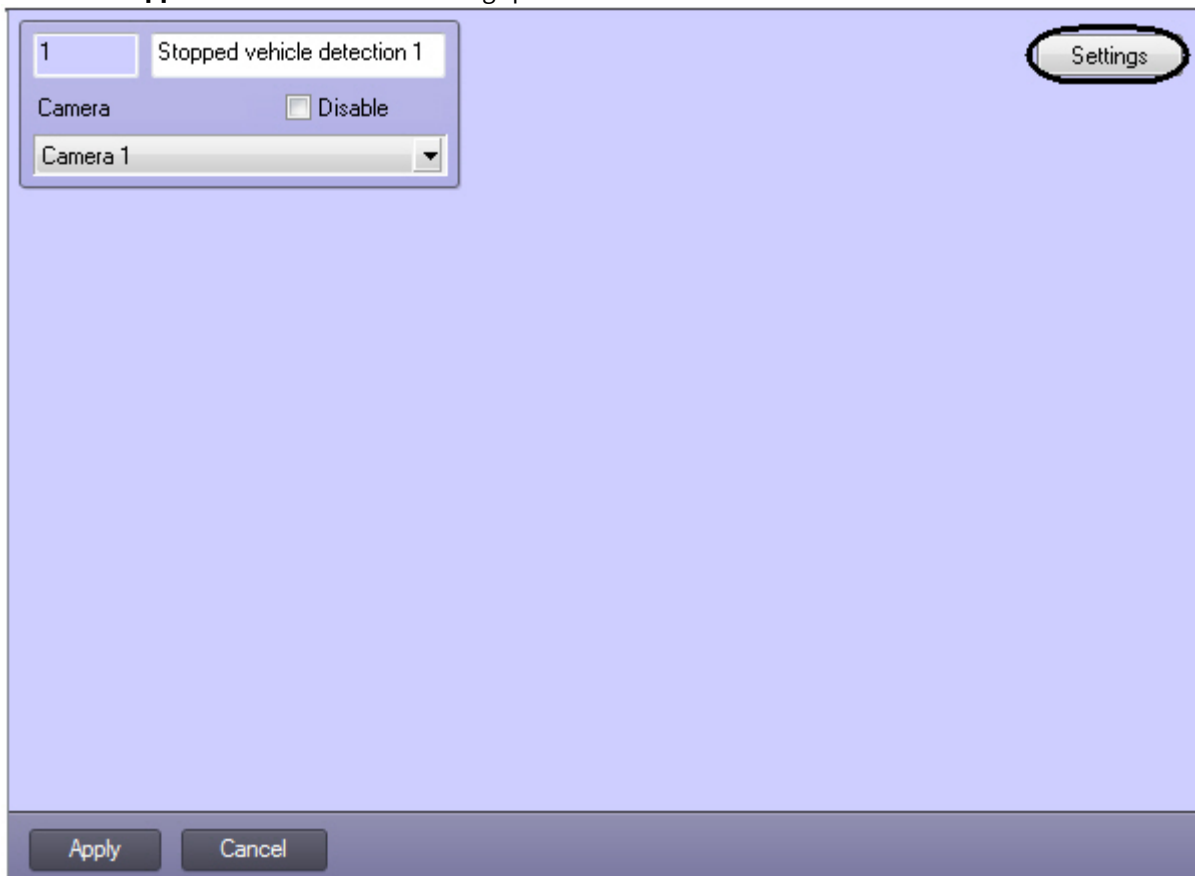
### 5.6.3.2 Configuring the Stopped vehicle detection module

The **Stopped vehicle detection** module can be configured using the **System settings** menu, under the **Hardware** tab, on the **Stopped vehicle detection** control panel, using the **Camera** settings.



The **Stopped vehicle detection** module is set up as follows:

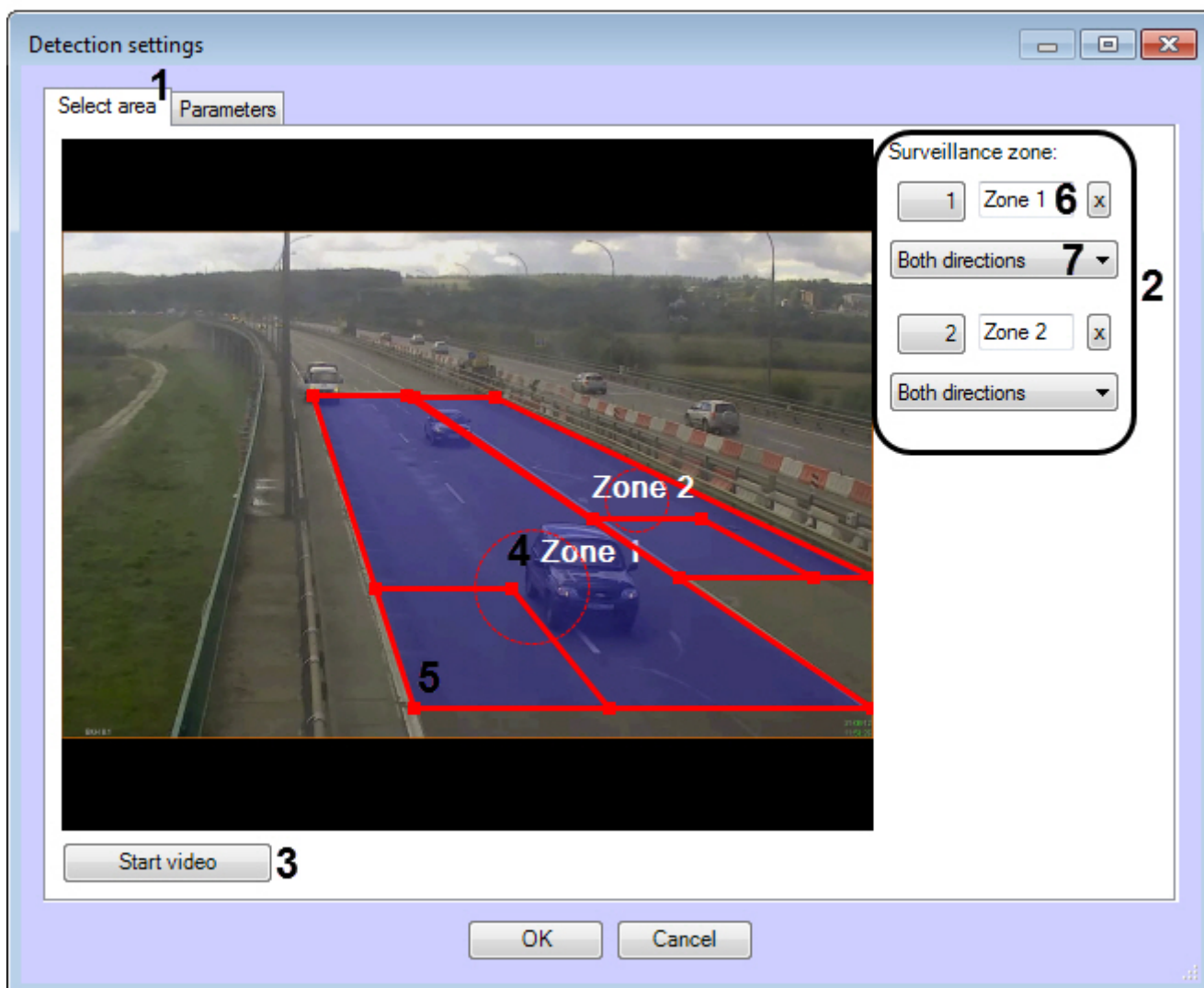
1. Go to the **Stopped vehicle detection** settings panel.



2. Click **Settings**. The **Detection settings** window will appear.
3. Configure the surveillance zones on the **Select area** tab (1). Several zones can be added (2). Detection tool triggering in each zone produces a separate event.

**Note.**

For each surveillance zone, both vehicle stop and vehicle leaving the zone are detected.



To add a surveillance zone, do the following:

- a. Click **Start video (3)** to capture the video image.
- b. To add a zone, click any mouse button on the video image.
- c. Set the required size, shape and position of the surveillance zone:
  - To change the zone size, use the nodal points of the zone;
  - To rotate the zone, drag it, while holding the mouse pointer close to the inner perimeter of the zone;
  - To move the zone around the frame of the video image, use a dotted line circle (4).
- d. Specify the approximate size of the vehicle by changing the size of the internal areas in the bottom left corner of the main zone (5). This field (5) is called the detection area.

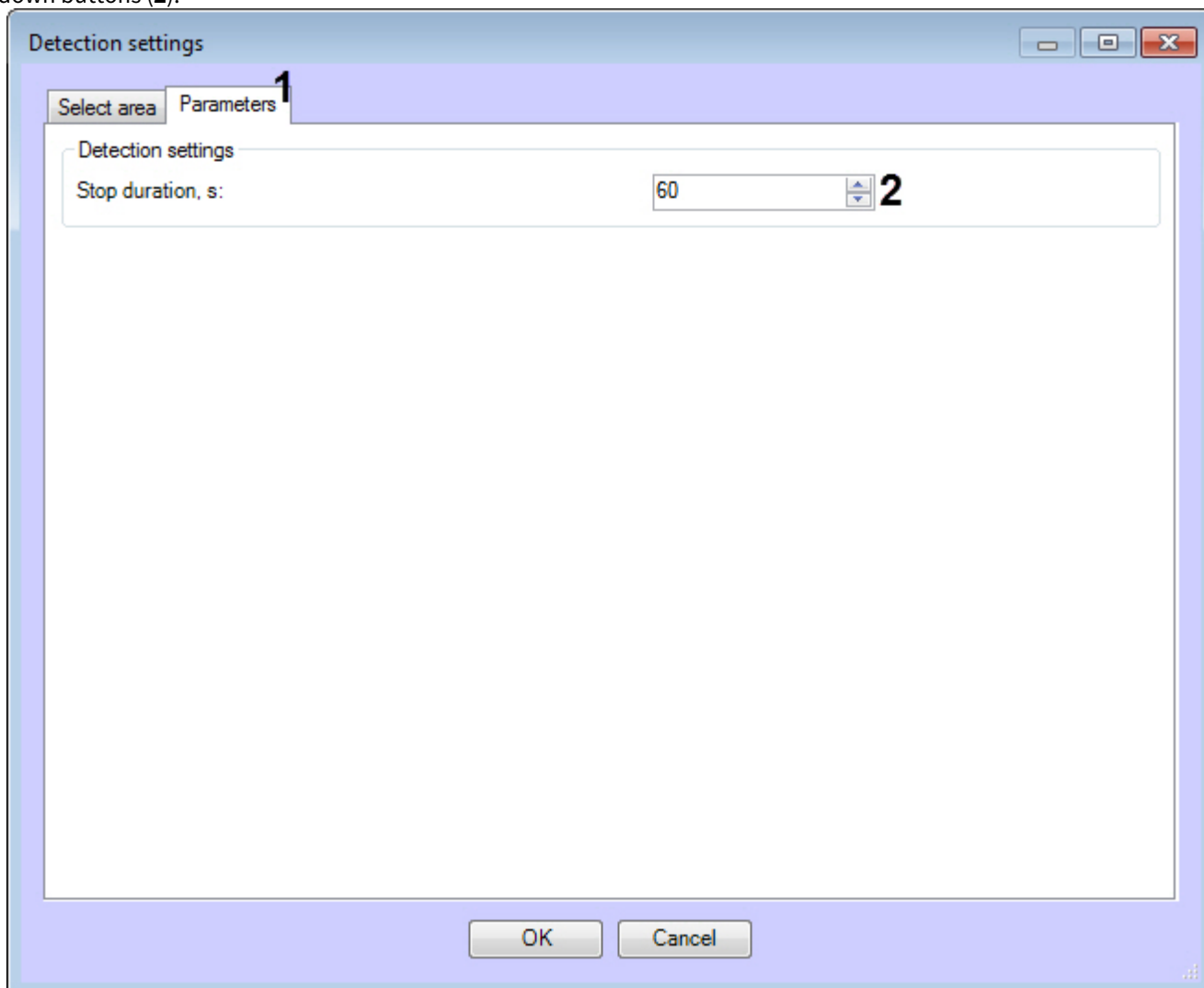
**Note.**

Each surveillance zone over its entire area is proportionally covered by detection areas of a given size. It is necessary to set the approximate dimensions of the vehicle so that each surveillance zone includes no more than 256 detection areas.

- To control the coverage of the zone by detection areas, use the debug window (see [Appendix 1. Debug window](#)).
- The stopped cars can be detected only in those parts of the surveillance zone, which are covered by detection areas.

- e. Enter zone name (6).
- f. Select the direction of the vehicles moving towards the camera for each zone (7).

- Go to the **Parameters** tab (1) and set the minimal stop duration in seconds in the **Stop duration, s** field using the up and down buttons (2).



- Click **OK**.

Configuring the **Stopped vehicle detection** module is completed.

## 5.7 Glow detection

### 5.7.1 Functionality of the Glow detection module

The **Glow detection** module is designed to carry out the following functions:

- Keeping track of light sources (lamps) in an observed area.
- Record events about recognizing of light sources insertion or elimination to the database.

### 5.7.2 Camera requirements for the Glow detection module

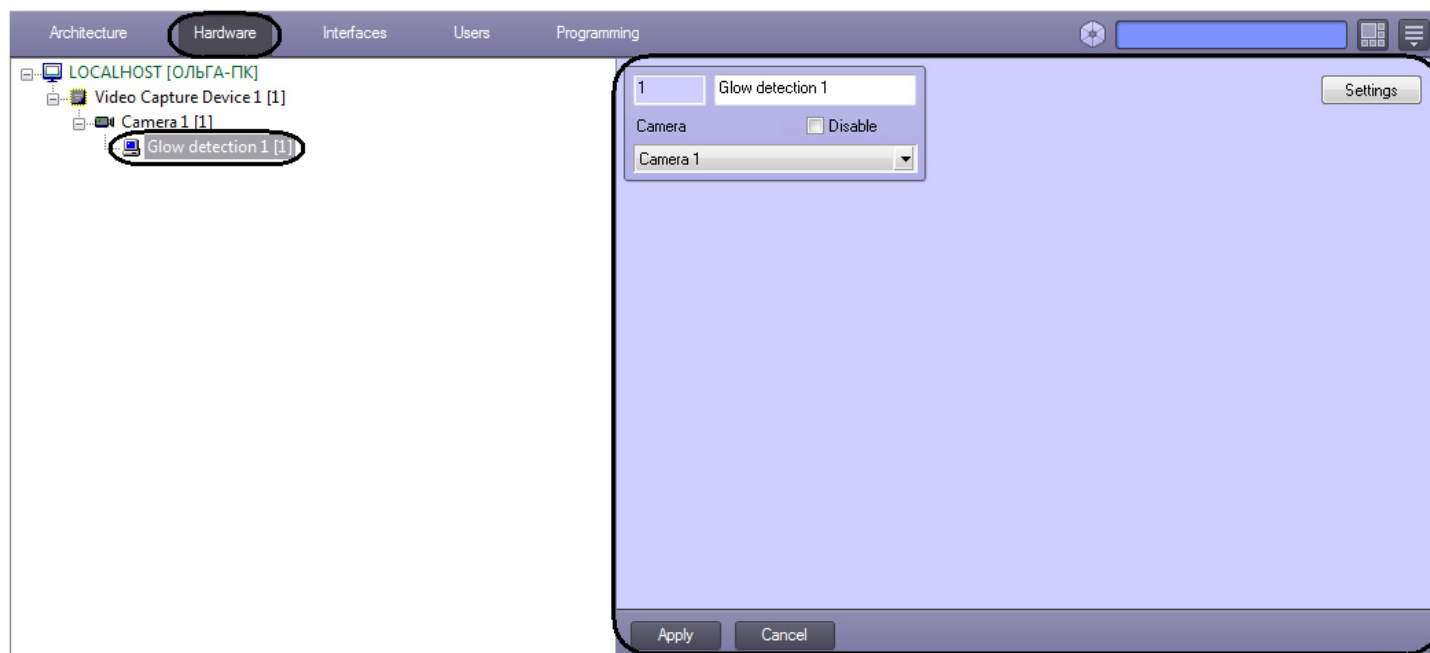
The requirements for the cameras that will work with the **Glow detection** module are listed in the following table.

Camera	<ul style="list-style-type: none"> <li>Resolution should be at least 320x240 pixels</li> <li>Fps: not less than 6</li> <li>Camera must be rigidly fixed.</li> </ul>
--------	---

<p>Scene and camera angle</p>	<ul style="list-style-type: none"> <li>• The camera is pointed to the area where all light sources are located (ideally, the optical axis of the camera is pointed strictly perpendicularly to this area).</li> </ul>
<p>Objects image</p>	<ul style="list-style-type: none"> <li>• Light sources in the video image should be visually distinguishable.</li> </ul>

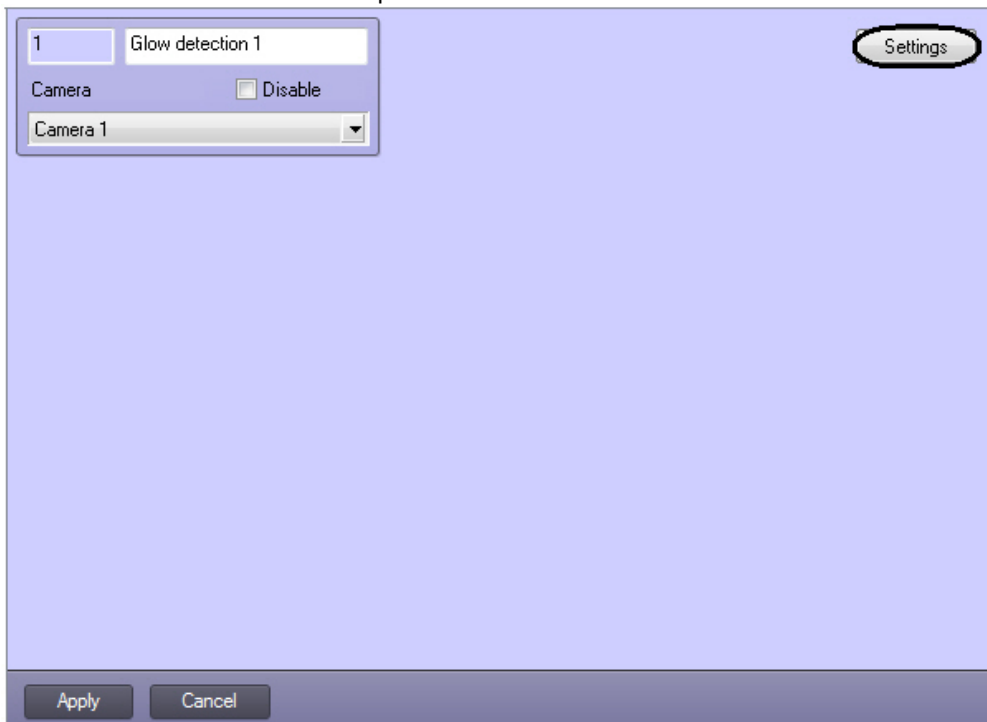
### 5.7.3 Configuring the Glow detection module

The *Glow detection* module can be configured using the **System settings** menu, under the **Hardware** tab, on the **Glow detection** control panel, using the **Camera** settings.

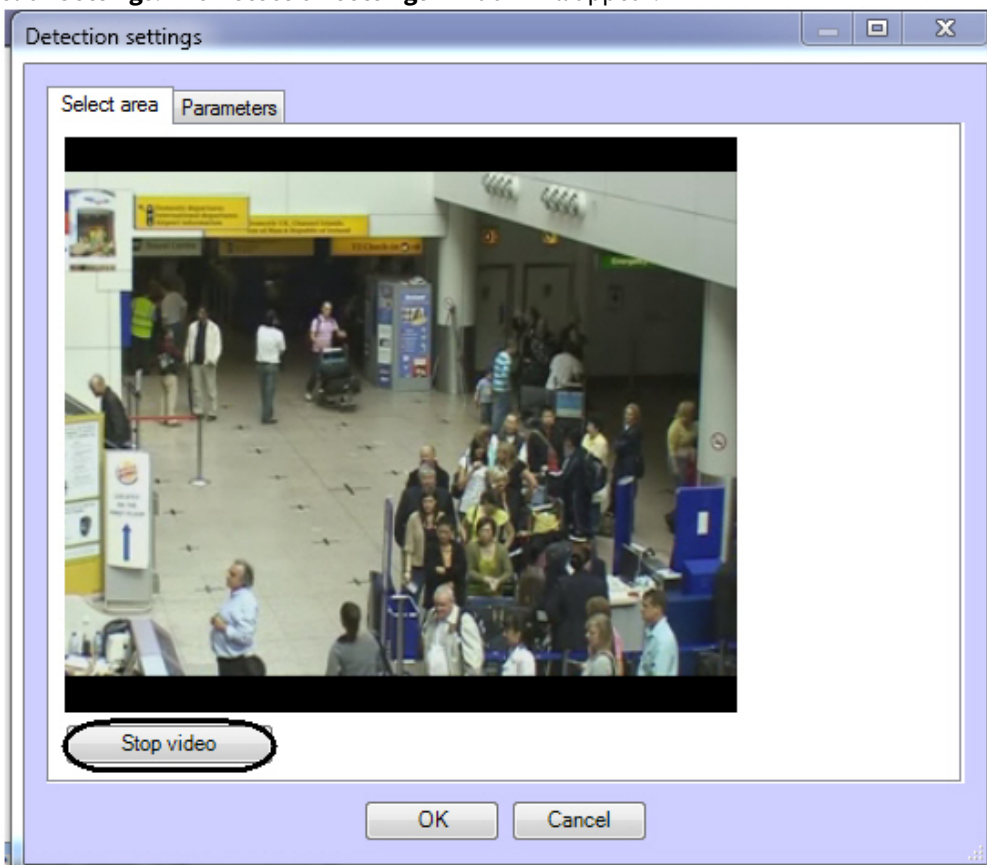


The *Glow detection* module is set up as follows:

1. Go to the **Glow Detection** control panel.

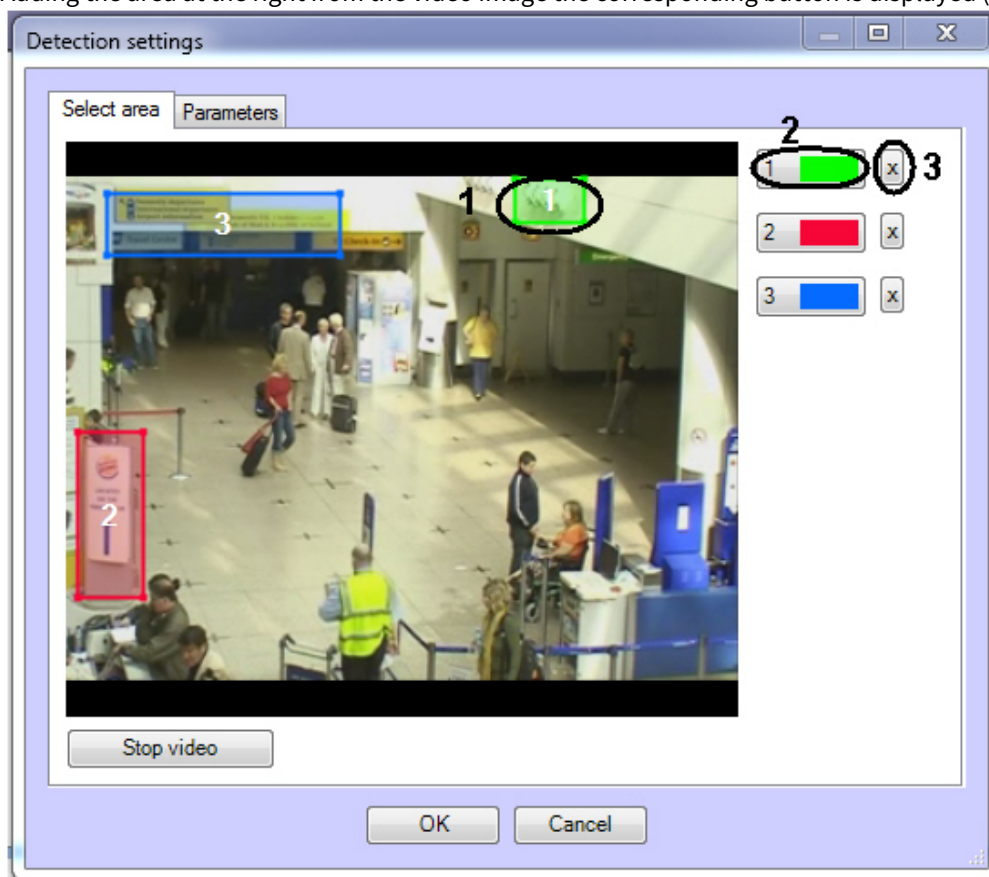


2. Click **Settings**. The **Detection settings** window will appear.



3. Specify the location of lights sources in the image which are to be tracked by detection:
  - a. Click **Stop video** to capture the video image.
  - b. On the captured video image specify areas to be analyzes (**1**). Click the left mouse button in the frame area and stretch it to the required size. The minimal allowed size of the analyzed area is 15x15 pixels. The maximum allowed size of the analyzed area is 200x200 pixels.

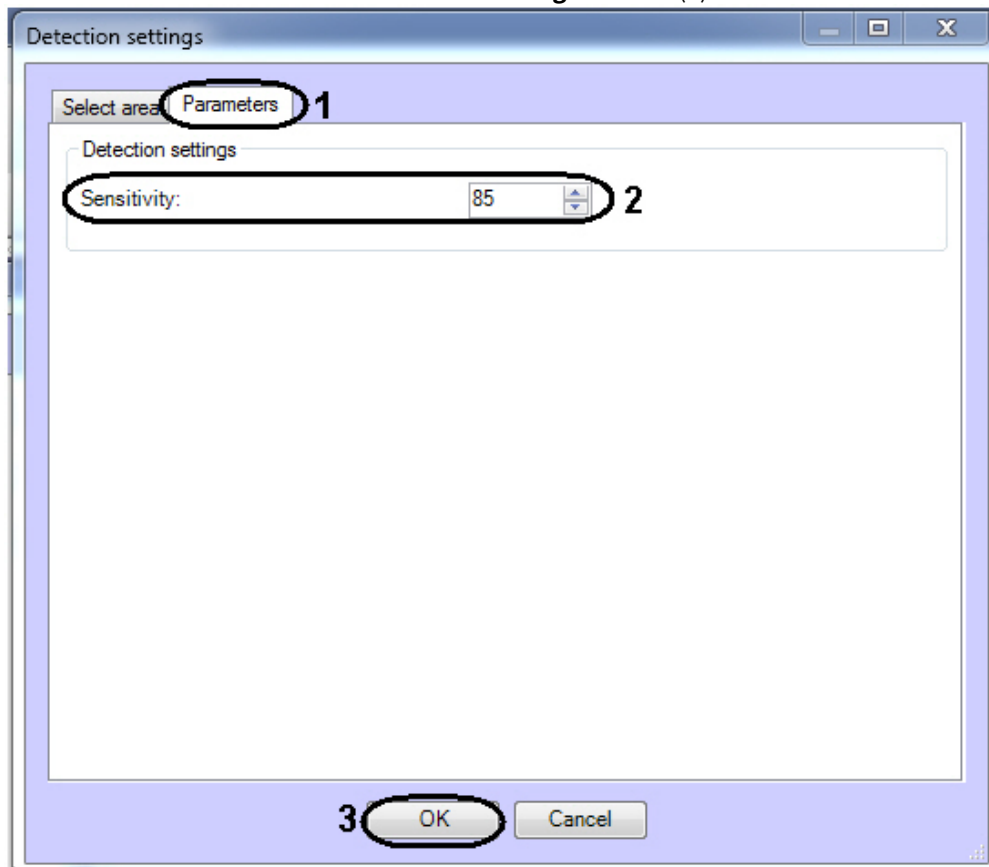
Areas of interest are numbered in the order of creation starting from 1. The number of analyzed area is not limited. Adding the area at the right from the video image the corresponding button is displayed (2).



- c. Specify the required size, shape and location of sectors in the area of interest moving their borders. Selecting the area take into account that the local change of illuminance and specular surface near lamps in the area of interest can cause the false detection triggering.
- d. To specify the area again click the button with its number in the list of areas and mark the area in the video image frame (2).
- e. Click the  button next to the **Area of interest** button (3).

4. Specify the glow detection sensitivity:

- a. Go to the **Parameters** tab in the **Detection settings** window (1).



- b. Using the **up-down** buttons enter the value of sensitivity parameter in the **Sensitivity** field (2). The optimal value is selected experimentally by testing detection on triggering in the required conditions. The value range is from 0 to 100. The more sensitivity the more possibility of false triggering. The less sensitivity, the more possibility of losing event.

5. Click the **OK** button (3).

Configuring the *Glow detection* module is complete.

## 5.8 Heat map detection

### 5.8.1 Functionality of the Heat map detection

The **Heat map detection** is designed to define zones of stopping and to estimate delay time of visitors in areas of interest.

### 5.8.2 Camera requirements for the Heat map detection module

The requirements for the cameras that will work with the **Heat map detection** module are listed in the following table.

<p>Camera</p>	<ul style="list-style-type: none"> <li>Resolution: 720x576 (CIF4), using of 360x288 (CIF1) is acceptable; oversize images are reduces until CIF4.</li> <li>Fps: not less than 6</li> <li>Color: analytics works with grey and color images.</li> <li>Camera must be rigidly fixed.</li> </ul>
---------------	---

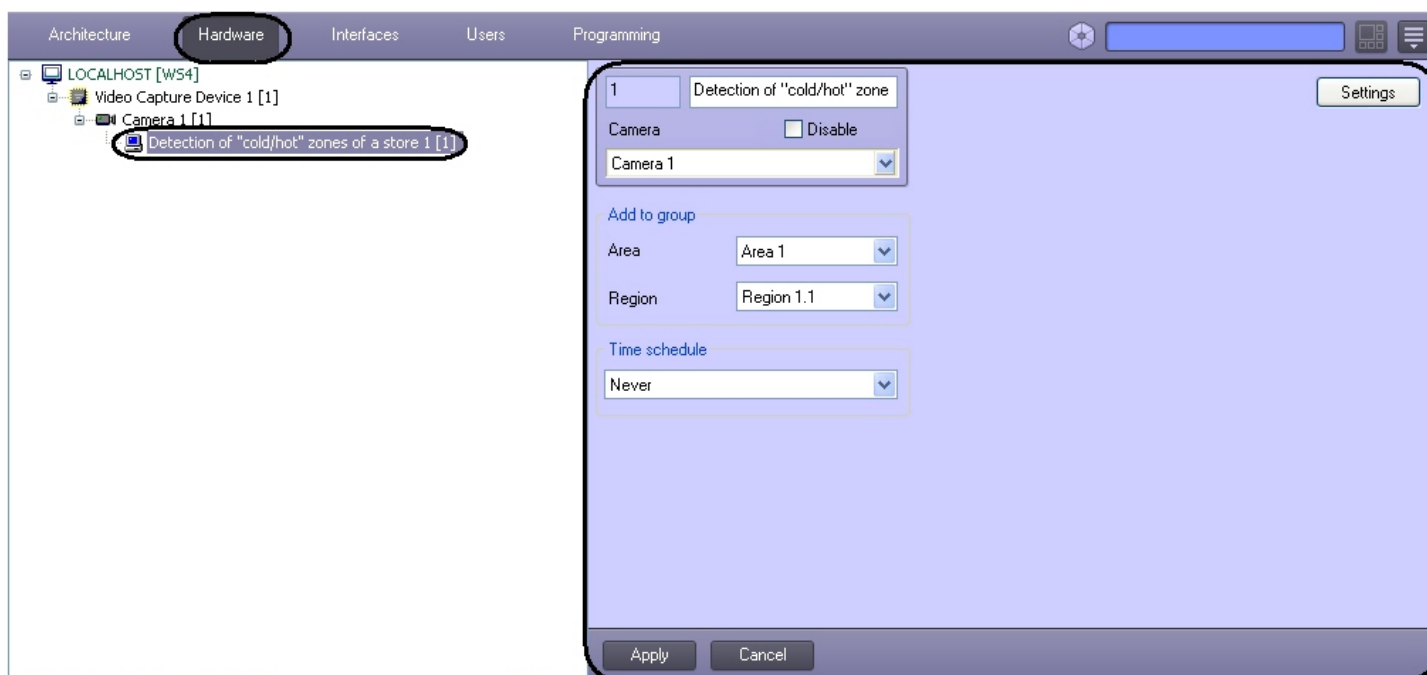
Lighting	<ul style="list-style-type: none"> <li>• The best working of detection is archived at medium lighting. In conditions of insufficient (night) or excessive (exposure) lighting, the quality of analytics can be reduced.</li> <li>• Sharp changes of lighting can lead to short-time invalid analytics working.</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>• The best position – camera "looks" to the scene vertically down. The better this requirement, the carefully the received estimation.</li> <li>• Sizes of camera field of view: 3x3m is minimal (6x6 people), 4x4m is optimal (8x8 people), 8x8m is maximal (16x16 people).</li> <li>• Background is static and is not changed sharply.</li> <li>• Analytics can work inappropriately on specular surfaces and in case of sharp shadows from moved objects.</li> <li>• Analytics can work inappropriately in case of in the camera field of view there are periodic movements of background objects (trees, working TV, etc.)</li> </ul>
Objects image	<ul style="list-style-type: none"> <li>• Image quality: the image is to be clear, without visible defects from reducing procedure.</li> <li>• Permissible size of a person (the area of a rectangular track around a person) as a percentage of the frame area: from 0.25% to 10%</li> </ul>

### 5.8.3 Configuring the Heat map detection module

**Attention!**

For correct operation of the *Heat map detection* module the installed [Analytics Pack subsystem](#) is required.

The **Heat map detection** module can be configured using the **System settings** menu, under the **Hardware** tab, on the **Heat map detection** control panel, using the **Camera** settings.



The **Heat map detection** module is set up as follows:

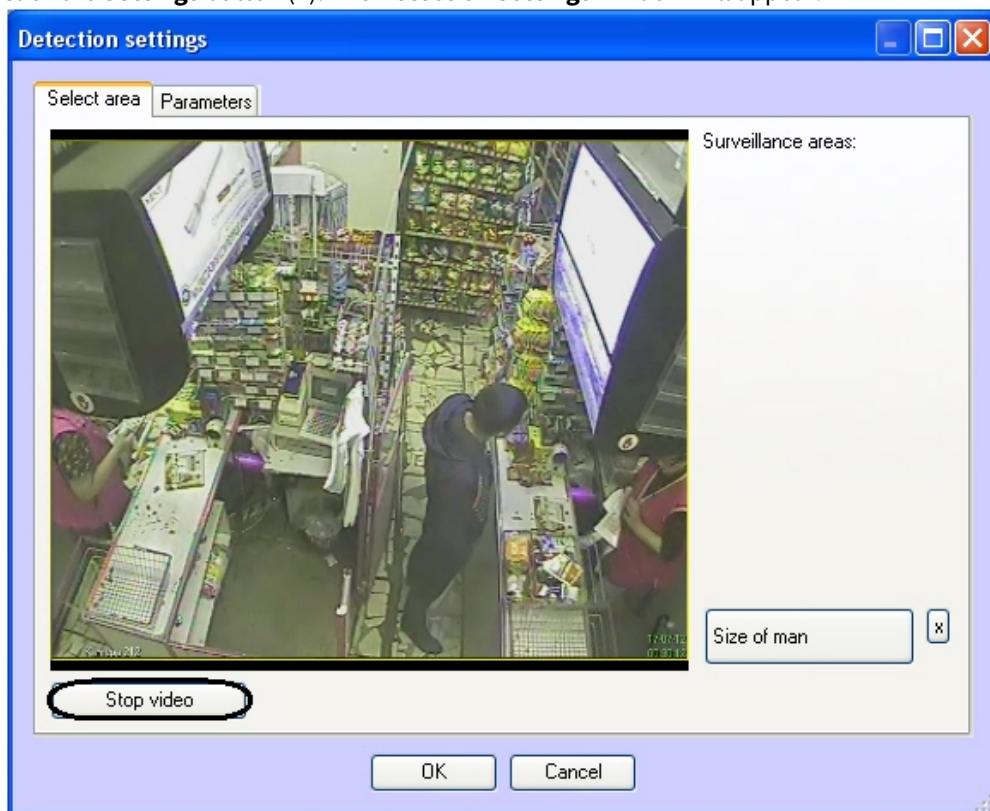
1. Go to the **Heat map detection** control panel.

2. From the **Area** drop-down list select the **Area** object to display the list of regions which belong to this area (**1**).
3. From the **Region** drop-down list select the **Region** object to which this detection is to be referred to (**2**).
4. From the **Time schedule** drop-down list select the corresponding time schedule during which the detection will work (**3**).

**Important!**

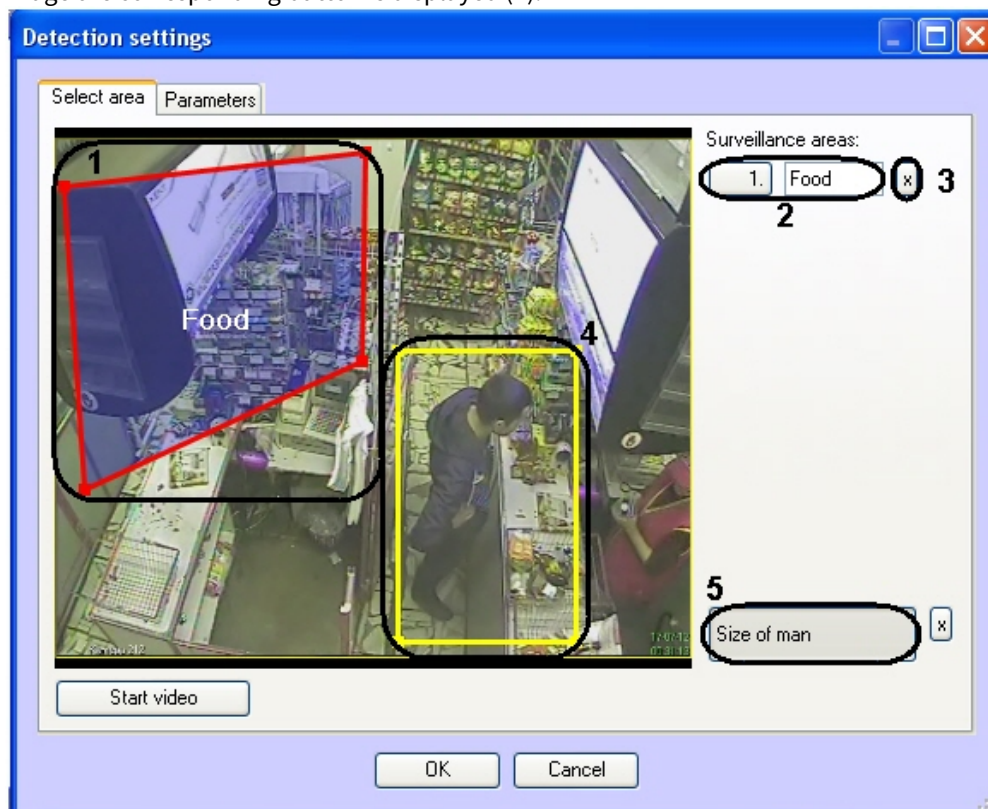
If no custom **Time zone** objects is created in the system, two time zones are available – **Always** and **Never**. The **Never** time zone is set by default, so the detection tool will not work if default settings are left unchanged.

5. Click the **Settings** button (4). The **Detection settings** window will appear.



6. Specify the area of interest and the approximate size of people in the video image:
- a. Click the **Stop video** button to capture the video image.
  - b. On the captured video image specify areas to be analyzed (1). To specify the area set the nodal points of interested area using the left mouse button. After setting the last nodal point click the right mouse button and closing the curve will perform automatically.  
Areas of interest are numbered in the order of creation starting from 1. Adding area at the right from the video

image the corresponding button is displayed (2).



c. To remove a selected area click the  button next to the **Area of interest** button (3).

7. Set the approximate person size as follows:

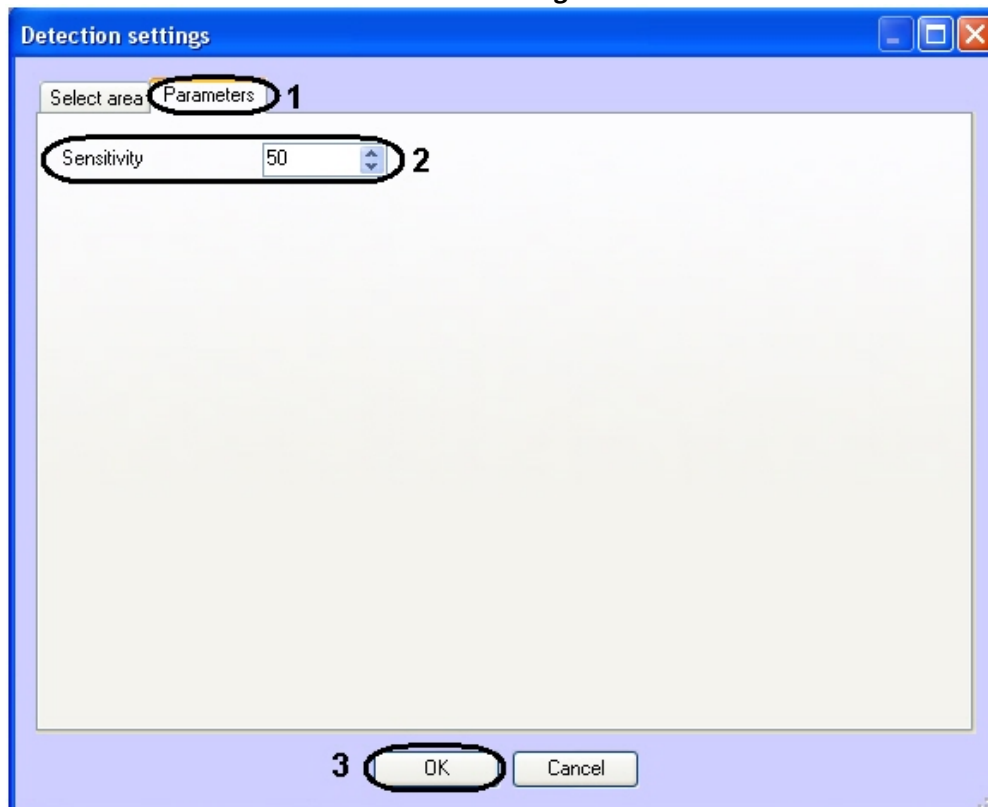
a. Click the **Size of man** button (5).

b. On the captured video image set the approximate person size. To do this, use the left mouse button to select a rectangular area (1).

c. To remove the person size click the  button next to the **Size of man** button.

8. Setting the module parameters:

- a. Go to the **Parameters** tab in the **Detection settings** window.



- b. Using the **up-down** buttons enter the value of sensitivity parameter in the **Sensitivity** field (2). The optimal value is selected experimentally by testing detection on triggering in the required conditions. The value range is from 0 to 100. The more sensitivity the more possibility of false triggering. The less sensitivity, the more possibility of losing event.

9. Click the **OK** button (3).

Configuring the Heat map detection module is completed.

## 5.9 Detection of moving against crowd flow

### 5.9.1 Functionality of the Detection of moving against crowd flow

The **Detection of moving against crowd flow** module is designed to detect objects moving in the video image in direction different from direction of movement the majority of same objects. The **Detection of moving against crowd flow** module performs the following functions:

1. Recognizing movement against a crowd.
2. Recording events about recognizing a movement against a crowd to the database.

### 5.9.2 Camera requirements for the Detection of moving against crowd flow module

The requirements for the cameras that will work with the **Detection of moving against crowd flow** module are listed in the following table:

Camera	<ul style="list-style-type: none"> <li>• Resolution should be at least 640x480 pixels</li> <li>• Fps not less than 6</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>• The camera “looks” to the scene vertically down</li> </ul>

Object images	<ul style="list-style-type: none"> <li>• People move in a video image from top to bottom or from bottom to top</li> </ul>
Lighting requirements	<ul style="list-style-type: none"> <li>• Objects should be visually separated from the background and from each other</li> </ul>

### 5.9.3 Configuring the Detection of moving against crowd flow module

#### 5.9.3.1 Behavior of the Detection of moving against crowd flow module

Operation of the detection of moving against crowd flow is controlled by the following parameters:

1. **Movement against a crowd** – direction reversed to direction in which a crowd is moved in the video image. Direction of crowd movement is called the right direction.
2. **Number of people in a crowd** – minimal number of people moved in direction of crowd movement in which detection should trigger.

Detection will not function in case of number of people in a crowd moved in right direction is less than the **Number of people in a crowd** parameter.

If number of people moved in right direction is more or equal to the **Number of people in a crowd**, then:

- detection functions if number of people moved against a crowd is from 1 to 2 included;
- detection doesn't function if number of people moved against a crowd is 3 or more;
- detection doesn't function if there is no people moved against a crowd.

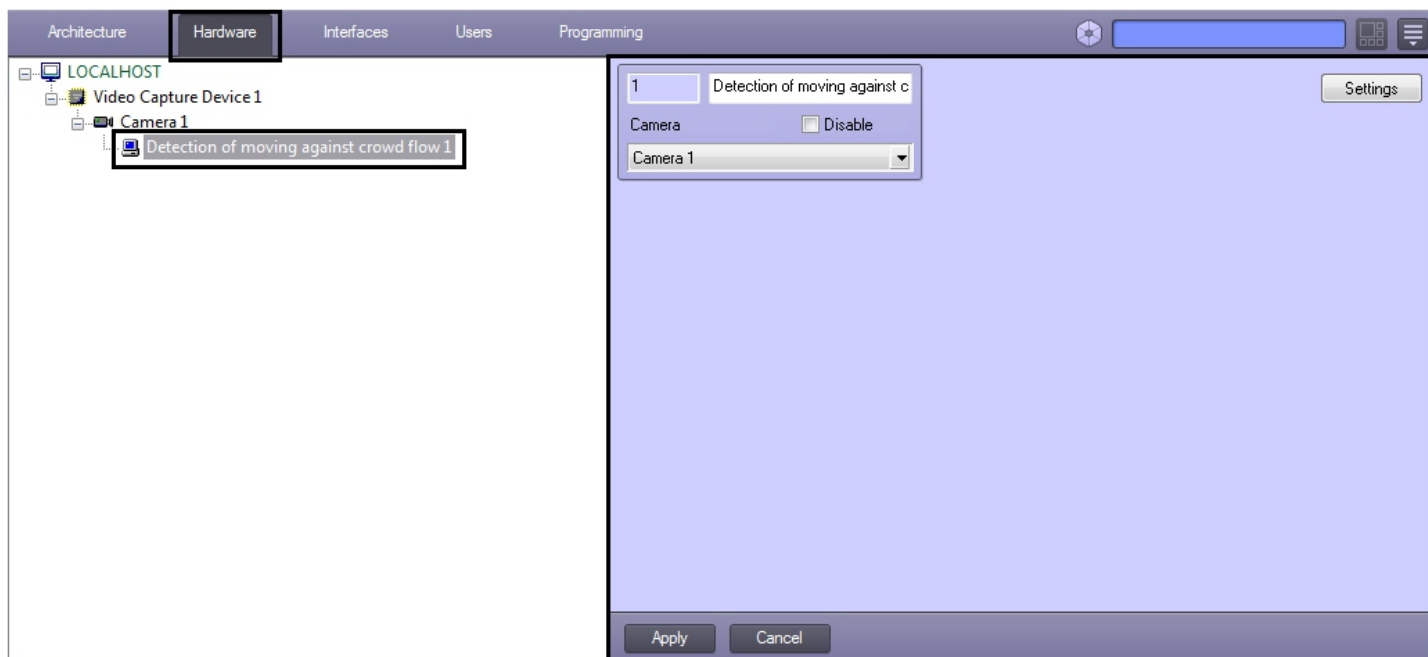
There will not new detection triggers while a person on which detection functioned is stayed in the monitored area.

**Example.** A steady flow of people goes against a crowd. In the monitored area only 1 or 2 persons move against a crowd. In this case only one trigger will happened when the first person will go against a crowd. New triggers will not happen until a flow moved against a crowd will not stopped.

If all people moved against a crowd are out of monitored area, i.e. in the monitored area there is only crowd moved in the right direction and new person moved against a crowd enter the area, then detection will function.

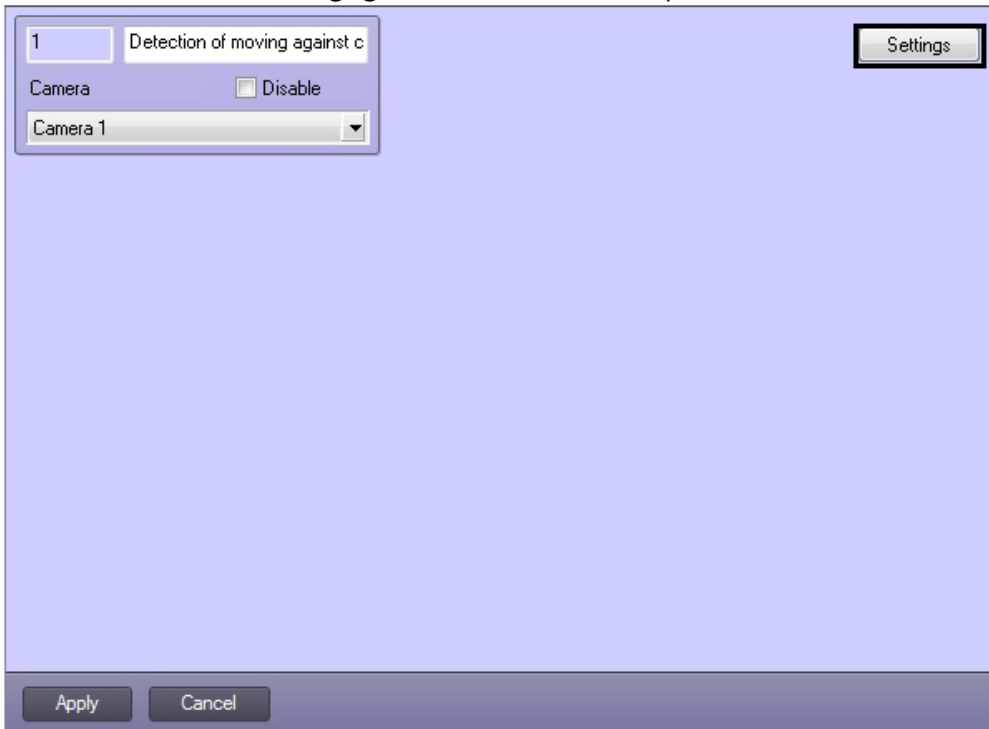
#### 5.9.3.2 Configuring the Detection of moving against crowd flow module

The *Detection of moving against crowd flow* module can be configured using the **System settings** menu, under the **Hardware** tab, on the **Detection of moving against crowd flow** object's control panel created on the basis of the **Camera** object.

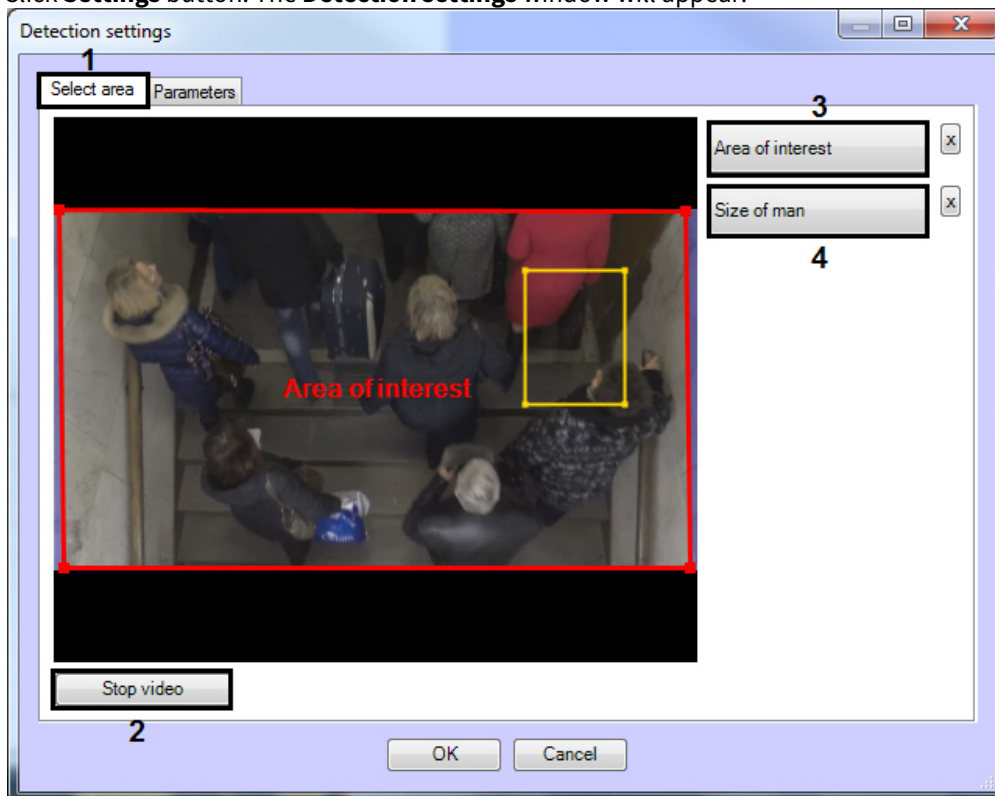


The *Detection of moving against crowd flow* module is set up as follows:

1. Go to the **Detection of moving against crowd flow** control panel.



2. Click **Settings** button. The **Detection settings** window will appear.



3. Specify the detection surveillance:
  - a. Go to the **Select area** tab (1).
  - b. Click the **Stop video** button to capture the video image (2).
  - c. Click the **Surveillance territory** button (3).
  - d. On the captured video image specify areas to be analyzed. To specify the area set the nodal points of interested area using the left mouse button. Area is considered to be specified when the last nodal point is consisted with the

first one. It is possible to add only one area. While attempting to add the second area the first one will be deleted. After area specifying the remaining part of video image will be darkened.

**Note.**

To remove the area click the  button next to the **Area of interest** button.

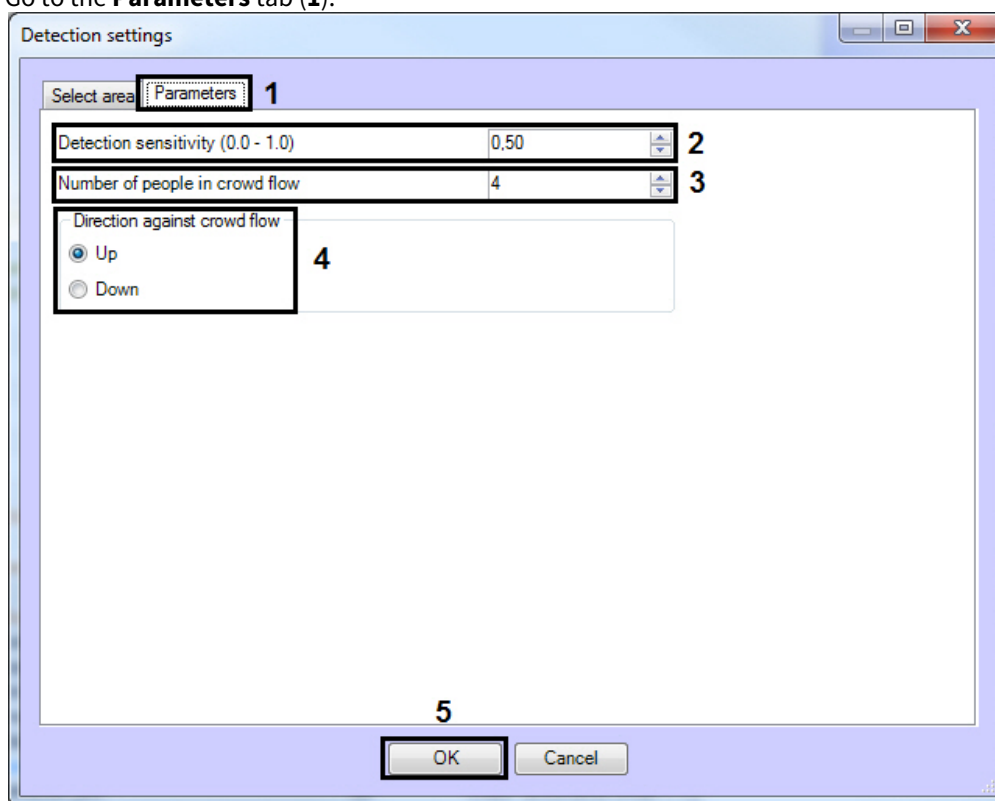
- e. Click the **Human size** button (4). Set the required person size. To do this, click the left mouse button on the captured video image and extend a rectangular area to required size.

**Note.**

To remove the area click the  button next to the **Size of man** button.

4. Specify parameters of detection of moving against crowd flow:

- a. Go to the **Parameters** tab (1).



- b. In the **Detector sensitivity** field enter the value of sensitivity parameter using up-down button (2). Optimal value of parameter is selected experimentally by testing the detection for triggers in required conditions. The range of values is from 0 to 1. The less detection sensitivity, the more probability of event missing.
- c. In the **Men in crowd** field specify the minimal number of people moved in direction of crowd movement in which detection should function (3).
- d. Set the **Direction against crowd** switch to the position, corresponding to the objects movement against a crowd on the video image (4).

5. Click the **OK** button (5).

Configuring the detection of moving against crowd flow module is complete.

## 5.10 Sweethearting detection

### 5.10.1 Functionality of the Sweethearting detection module

The **Sweethearting detection** module is designed to prevent cases of intentional employee theft by scan avoidance at the cash register (sweethearting). The module works together with *POS-Intellect* software and *Intellect Web Report System* subsystem.

The **Sweethearting detection** module provides the following functions:

1. Recognizing the events of scanning at cash registers in real-time video.
2. Recording events of successful scanning to the event log.
3. Recording events of successful scanning to the database.

#### Note

Theft facts are displayed in the *Intellect Web Report System* in the **Sweethearting** report. When creating the report, events from *POS-Intellect* and the **Sweethearting detection** module are compared. If the detection tool recorded the carrying the goods, but events from *POS-Intellect* were not recorded at that time, then the theft occurred. Other cases are considered normal. For more information about how to configure and work with *POS-Intellect* and *Intellect Web Reports System*, see *POS-Intellect. Administrator's Guide* and *Intellect Web Report System. User guide* (the latest versions of these documents are available in the [AxxonSoft documentation repository](#)).

#### Note

The real theft detection probability is 50% in order to avoid frequent false triggerings of the detection tool. Though, multiple theft attempts will be detected with a 90% probability. Mainly theft detection probability depends on whether the requirements given in the [Camera requirements for the Sweethearting detection module](#) section are met.

### 5.10.2 Camera requirements for the Sweethearting detection module

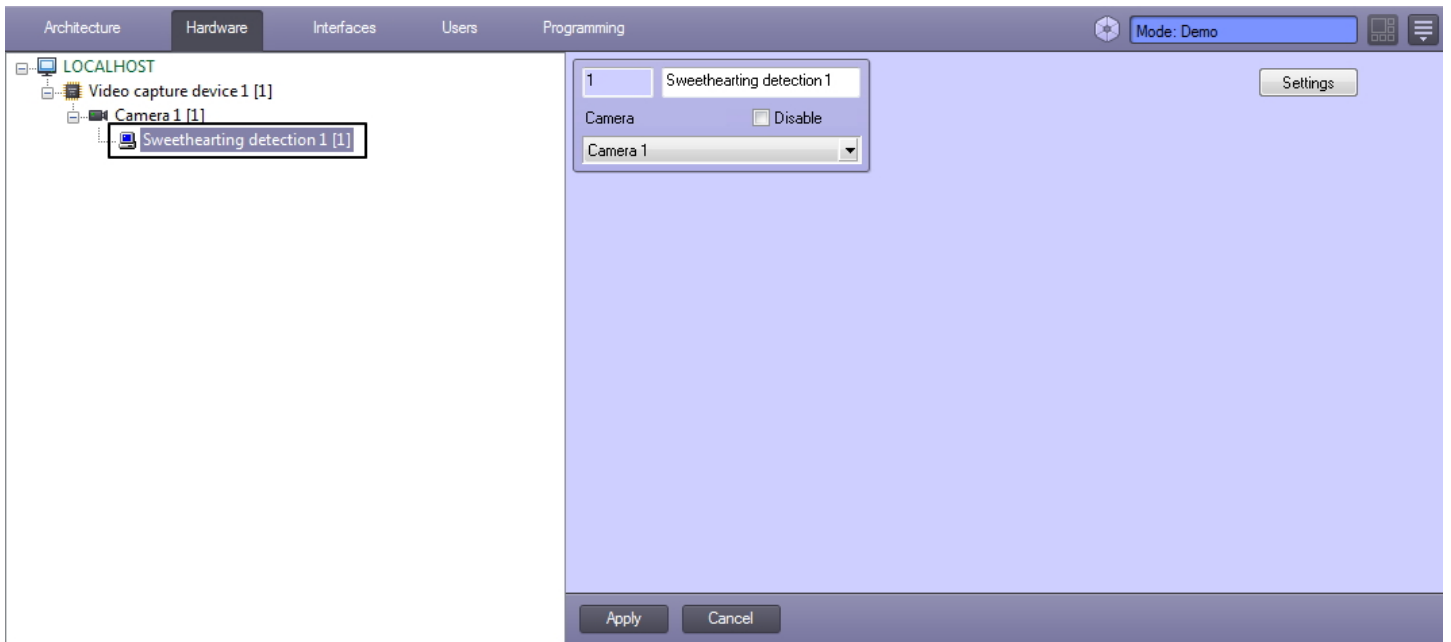
The requirements for the cameras that will work with the **Sweethearting detection** module are listed in the following table:

Camera	<ul style="list-style-type: none"> <li>• Resolution should be at least 1920x1080 pixels</li> <li>• Fps not less than 12</li> <li>• Only color cameras</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>• The angle of the CCTV camera shall be set so that the cashier's hands are clearly visible. The control area (where the cashier holds hands with the goods in front of the reader) should be from 150x150 to 250x250 pixels.</li> </ul>
Object images	<ul style="list-style-type: none"> <li>• The area of interest of the detection tool (see <a href="#">Configuring the «Sweethearting detection» module</a>) should be set in a way that the initial position of the goods is outside this area. The goods should be carried along the whole given area (in case of vertical or horizontal location of the area – from one side to the other, and in the case of the area at an angle – from one diagonal to another).</li> </ul>
Lighting requirements	<ul style="list-style-type: none"> <li>• Objects should be visually separated from the background and from each other</li> </ul>

For video cards requirements used for recognition channels operation, see [Selecting types of detection tools](#).

### 5.10.3 Configuring the Sweethearting detection module

The **Sweethearting detection** module is configured on the settings panel of the **Sweethearting detection** object created under the **Camera** object on the **Hardware** tab of the **System settings** dialog box.



The **Sweetheating detection** module is configured as follows:

1. Go to the settings panel of the **Sweetheating detection** module.
2. Click the **Settings** button (1).



The **Detection settings** window will open.



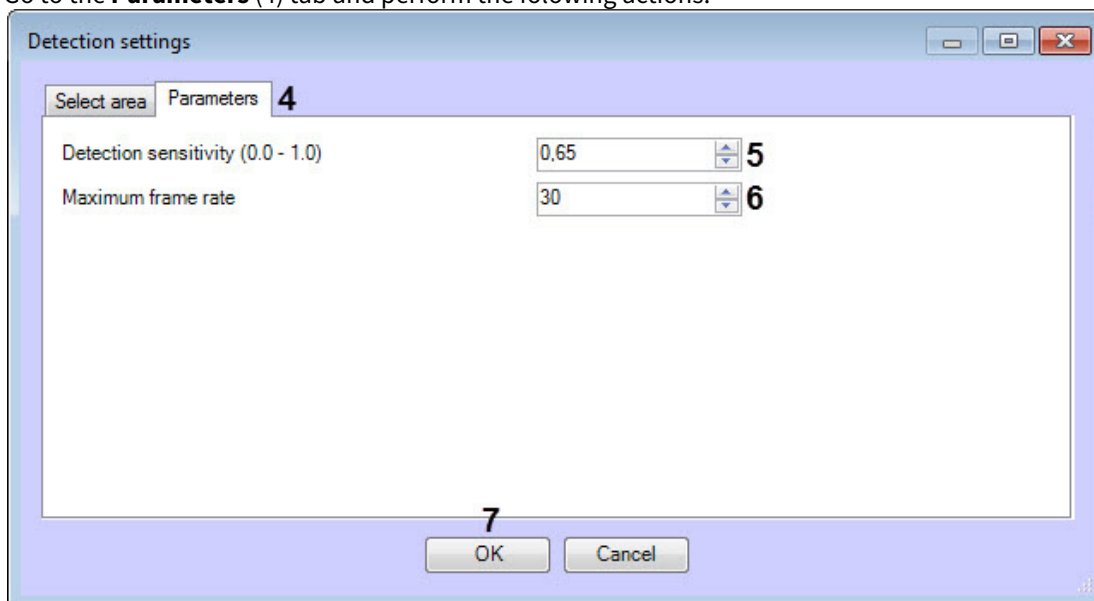
3. Specify the area of interest of the detection:
  - a. Go to the **Select area** tab (1).
  - b. Click **Stop video** to capture the video image (2).
  - c. Click **Area of interest** (3).
  - d. Specify the area of interest in the captured video image to be analyzed. The selected area must comply with [Camera and camera settings requirements for the Sweetheating detection module](#) .

**Note**

Only one area can be specified. If the second area is specified, then the first area will be deleted.

To remove a selected area, click the  button next to the **Area of interest** button.

4. Go to the **Parameters** (4) tab and perform the following actions:



5. Specify the detection sensitivity in the range from 0.0 to 1.0 in the **Detection sensitivity (0.0 - 1.0)** parameter (5).

**Note**

By default the 0.65 value is set in the **Detection sensitivity (0.0 - 1.0)** parameter.  
The value is found experimentally.

6. Specify the number of frames processed per second by the detection in the **Maximum frame rate** parameter (6).

**Note**

2-3 fps are enough for proper operation of the **Sweetheating detection** module.

7. Click **OK** (7) to save changes and return to the settings panel of the **Sweetheating detection** object.

**Note**

To return to the control panel of the **Sweetheating detection** without saving changes, click **Cancel**.

8. Click the **Apply** button (2) on the **Sweetheating detection** control panel.

The **Sweetheating detection** module is now configured.

## 5.11 Barcode detection

### 5.11.1 Functionality of the Barcode detection

The **Barcode detection** is designed to define barcodes or QR-codes in areas of interest.

### 5.11.2 Camera requirements for the Barcode detection module

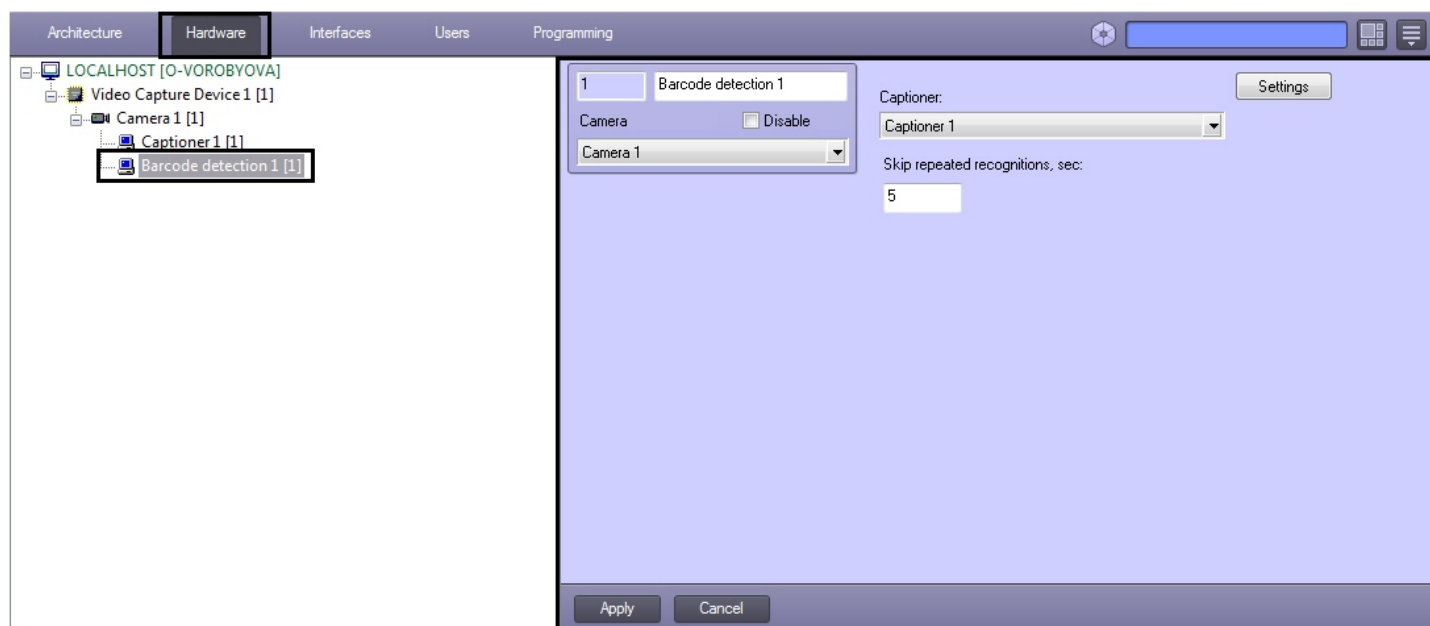
The requirements for cameras that will work with the **Barcode detection** module are given in the table below.

Camera	<ul style="list-style-type: none"> <li>The resolution depends on the barcode size in the frame. If the barcode size takes half the frame, then a resolution of 260 pixels wide is enough.</li> <li>FPS rate: 6</li> </ul>
--------	---

Lightning	<ul style="list-style-type: none"> <li>The image should be clear, lines should be visually separable from each other</li> </ul>
Scene and camera view	<ul style="list-style-type: none"> <li>The angle between the optical axis of the camera and the plane of the barcode should be not more than 15°</li> </ul>
Images of objects	<ul style="list-style-type: none"> <li>Maximum width and height of the detected barcode – 65536 pixels</li> <li>The area of the detected barcode is not less than 1296 pixels</li> <li>Each of barcode sides is not less than 10 pixels</li> </ul> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p><b>Note.</b></p> <p>For example, if the barcode height is 10 pixels, than its width should be not less than 130 pixels. Vice versa, if the barcode width is 10 pixels, than its height should be not less than 130 pixels</p> </div>

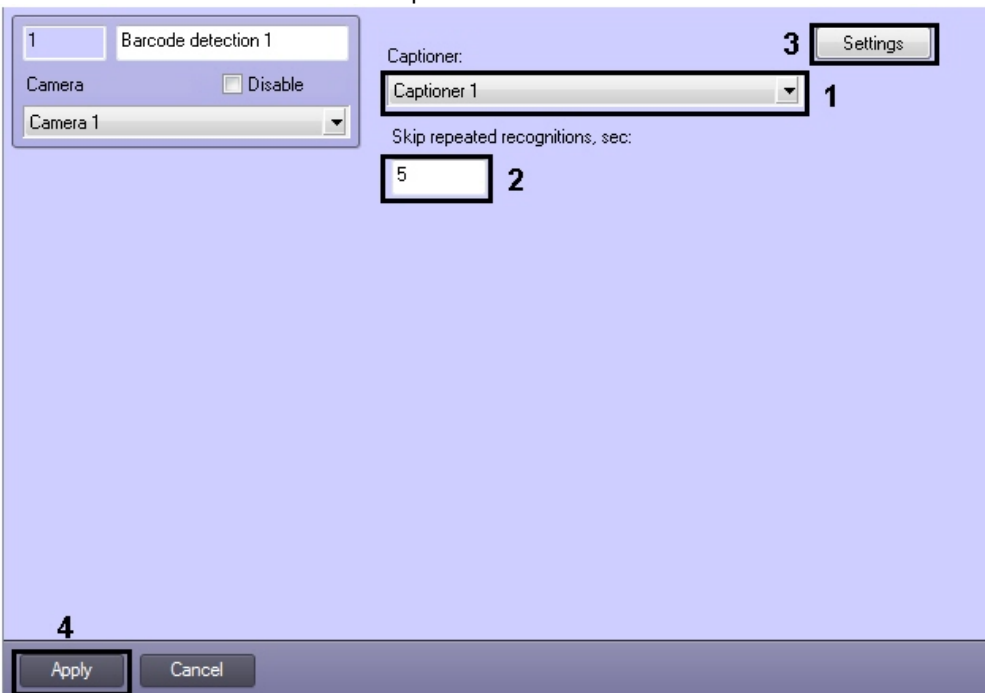
### 5.11.3 Configuring the Barcode detection module

The **Barcode detection** module can be configured using the **System settings** menu, under the **Hardware** tab, on the **Barcode detection** control panel, using the **Camera** settings.



The **Barcode detection** module is set up as follows:

1. Go to the **Barcode detection** control panel.

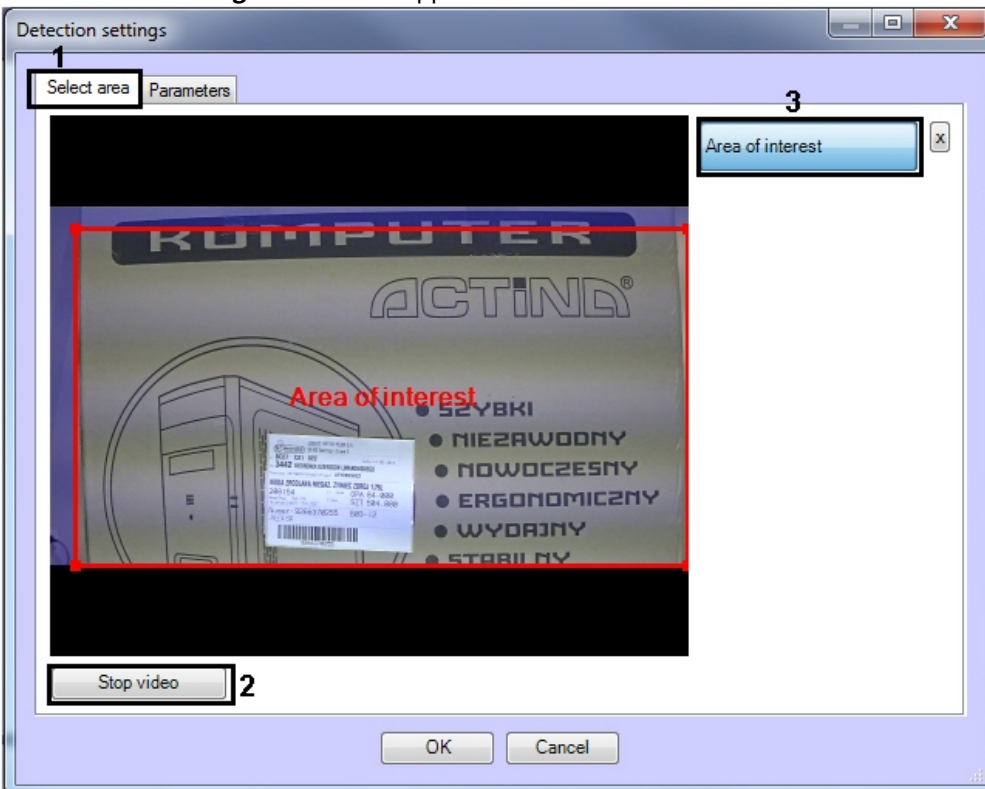


2. From the **Captioner:** drop-down list select the captioner with the help of which result will be displayed in the monitor (1)
3. In the **Skip repeated recognitions, sec:** field enter the time in seconds in which the repeated code is recognized (2).

**Note.**


If there are some different barcodes or QR-codes one after another, then recognition is performed instantly. If there are some repeated barcodes then new result will be displayed after time period specified in the settings.

4. Click **Settings (3)**.  
The **Detection settings** window will appear.



5. Specify the detection surveillance area:
  - a. Go to the **Select area** tab (1).
  - b. Click the **Stop video** button to capture the video image (2).
  - c. Click the **Area of interest** button (3).
  - d. On the captured video image specify areas to be analyzed. It is possible to add only one area. While attempting to add the second area the first one will be deleted. After area specifying the remaining part of video image will be darkened.

**Note.**

To remove the area click the  button next to the **Area of interest** button.

- e. Click **OK** to save changes and return to the control panel of the **Barcode detection** (4).

**Note.**

To return to the control panel of the **Barcode detection** without saving changes, click **Cancel**.

6. On the **Barcode detection** control panel, click **Apply**.

Configuring the **Barcode detection** module is complete.

## 5.12 Train detection

### 5.12.1 Functionality of the Train detection module

The **Train detection** module is designed to carry out the following functions:

1. Recognizing of train presence/absence in the monitored area.
2. Recording events of train appearance in the monitored area to the database.
3. Recording events of train disappearance from the monitored area to the database.

### 5.12.2 Camera requirements for the Train detection module

The requirements for the cameras that will work with the **Train detection** module are listed in the following table.

Camera	<ul style="list-style-type: none"> <li>• Resolution should be at least 640x480 pixels.</li> <li>• Fps: not less than 6</li> <li>• Color: analytics works with grey and color images.</li> <li>• Camera shaking should not result in image offsets greater than 1% of the frame size.</li> </ul>
Lighting	<ul style="list-style-type: none"> <li>• Medium lighting. In conditions of insufficient (night) or excessive (exposure) lighting, the quality of analytics can be reduced.</li> <li>• No sharp changes of lighting.</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>• Camera is to be directed in the line of the railway or as close to it as possible.</li> <li>• Background is mostly static and is not changed sharply.</li> </ul>

Objects image

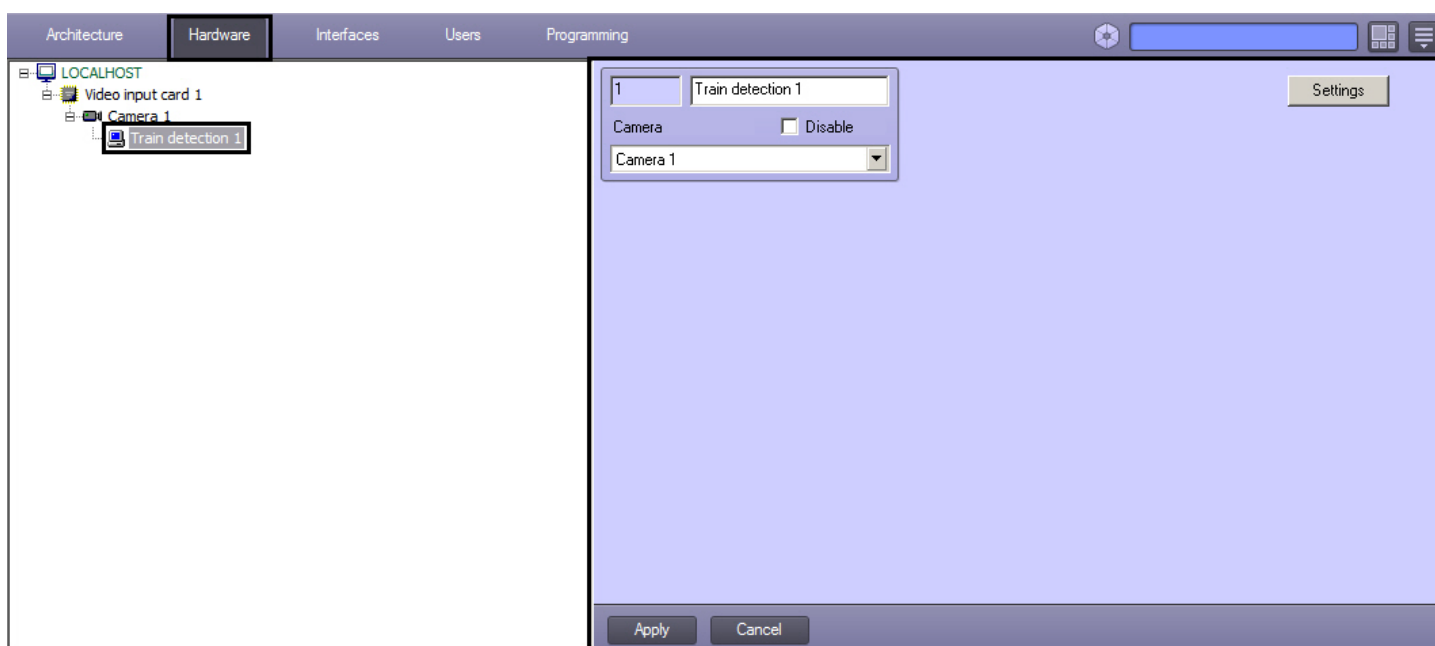
- At the time the detector was started, there was no train in FoV.
- Minimum overlapping of moving objects with the static objects in the scene (columns, trees, etc.).
- Analytics may not work correctly on reflective surfaces and if there are sharp shadows from moving objects.
- The width and height of objects in the image should not exceed 75% of the frame size.
- The speed of objects in the image should not be less than 1 pixel per second.

### 5.12.3 Configuring the Train detection module

**Note.**

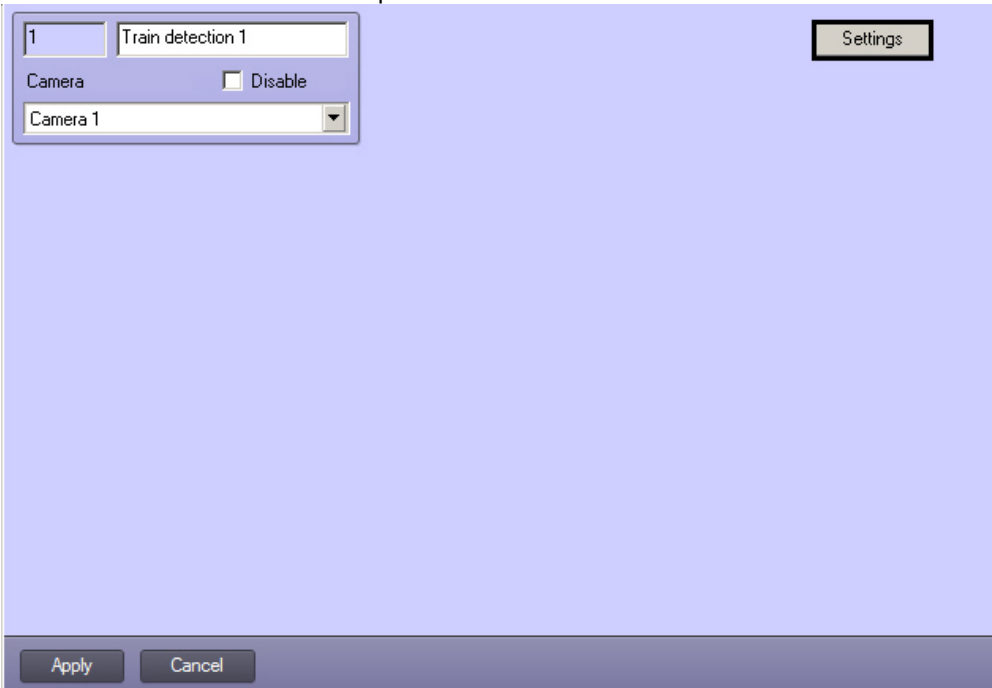
It is not recommended to create more than 4 **Train detection** objects for correct operation of the *Train detection* module.

The **Train detection** module can be configured on the settings panel of the **Train detection** object created on the basis of the **Camera** object on the **Hardware** tab of the **System settings** dialog window.

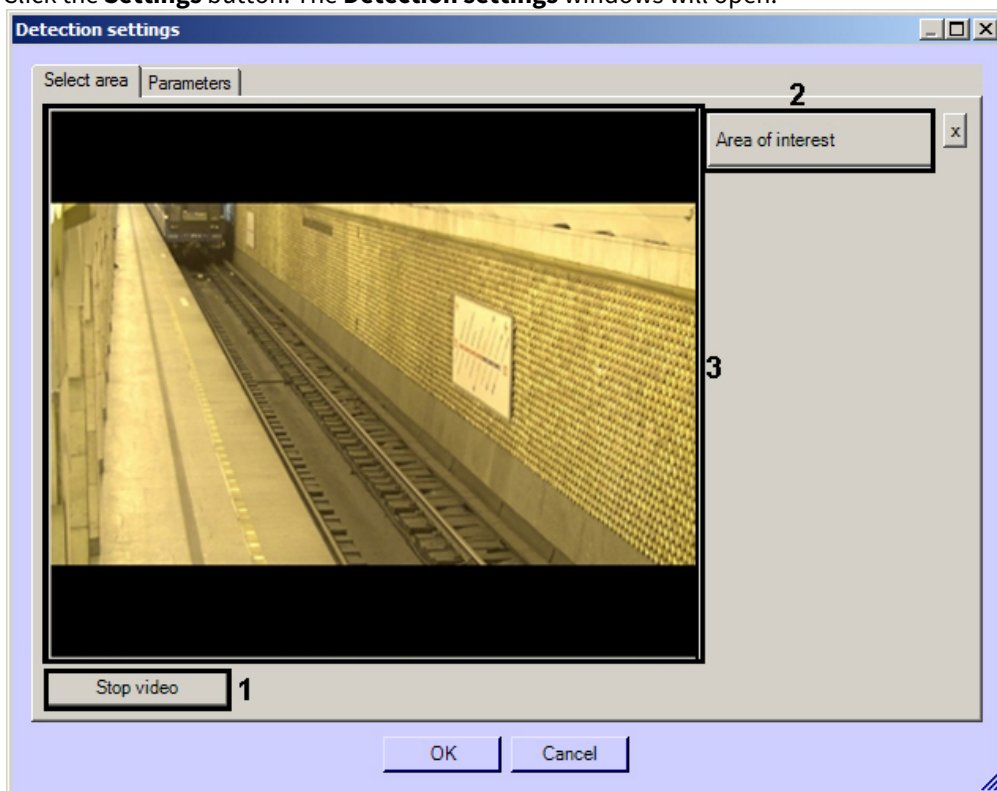


The **Train detection** module is configured as follows:

1. Go to the **Train detection** control panel.



2. Click the **Settings** button. The **Detection settings** windows will open.




3. Specify the detection surveillance area on the video image:

- a. Click the **Stop video** button to capture the video image (1).



- b. Click the **Area of interest** button (2).
- c. On the captured video image (3) sequentially specify nodal points of area to be analysed by clicking the left mouse button (1). It is possible to add only one area. While attempting to add the second area the first one will be deleted. After area specifying the remaining part of video image will be darkened.

**Note.**

To remove the area click the  button next to the **Area of interest** button.

**Note.**

The area of interest is to be specified in such a way as to be no any motion in it except of the train motion.

- d. Click the **OK** button to save changes and return to the control panel of the **Train detection** (4).

**Note.**

To return to the control panel of the **Train detection** without saving changes, click **Cancel**.

- 4. Click **Apply** on the **Train detection** control panel.

Configuring the **Train detection** module is complete.

## 5.13 Fire detection and Smoke detection

### 5.13.1 Functionality of the Fire detection and Smoke detection modules

#### Attention!

Unlike standard fire/smoke detection systems, the "Fire detection" and "Smoke detection" modules face many issues related to the scene and background in the video image. Thus, we cannot guarantee 100% smoke/fire detection. The smoke and fire detection tools are meant to increase the likelihood of fire/smoke detection. However, there may be both false alarms and failures to detect actual cases of fire/smoke in the camera's FoV.

The **Fire detection** and **Smoke detection** modules is designed to carry out the following functions:

1. Recognizing fire/smoke in the specified area of video image.
2. Recording events of fire recognition to the database.

### 5.13.2 Camera requirements for the Fire detection and Smoke detection modules

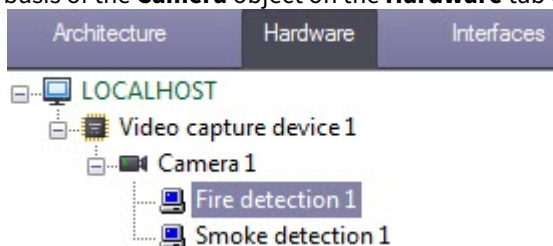
The requirements for cameras that will work with the **Fire detection** and **Smoke detection** modules are shown in the following table.

Camera	<ul style="list-style-type: none"> <li>• Resolution should be at least 640x480 pixels</li> <li>• Number of frames: the response rate of the detection module depends on the number of frames per second, by default, it is 1 per every 10 seconds (in most cases, it is enough to detect a fire/smoke that lasts more than 1 minute)</li> <li>• It is recommended to use color cameras. With black and white cameras, the recognition quality may be much worse</li> </ul>
Object images	<ul style="list-style-type: none"> <li>• The area of fire/smoke (the area of a rectangular track) should be at least 10% of the frame area</li> </ul>
Lightning	<ul style="list-style-type: none"> <li>• Fire/smoke should be visually separated from the background.</li> </ul>

### 5.13.3 Configuring the Fire detection and Smoke detection modules

#### Camera requirements for the Fire detection and Smoke detection modules

- The **Fire detection** module can be configured on the settings panel of the **Fire detection** object created on the basis of the **Camera** object on the **Hardware** tab of the **System settings** dialog window.
- The **Smoke detection** module can be configured on the settings panel of the **Smoke detection** object created on the basis of the **Camera** object on the **Hardware** tab of the **System settings** dialog window.



The **Fire detection** and **Smoke detection** modules are configured similarly and as follows:

1. Go to the appropriate detector object's control panel.

2. In the **Skip repeated recognitions, sec** field (1) enter the time period in seconds during which the repeated recognitions of the ignition/smoke will be ignored.

**Note**

For example, if 5 seconds is set for the **Time in seconds between processed frames** parameter, and there is a fire/smoke on the image during a longer period of time, then in order to avoid receiving messages of ignition/smoke from the detector every 5 seconds, set a higher value for the **Skip repeated recognitions, sec** parameter (for example, 10 seconds). In this case, only one message about the ignition/smoke will be displayed, and all the repeated recognitions will not be displayed in the *Event viewer* until the time interval between the recognitions of the ignition/smoke becomes longer than 10 seconds.

3. In the **Detection sensitivity: 0..100** field (2) enter the sensitivity of detection – integer value from 0 to 100. This parameter has an impact on sufficient probability of fire/smoke recognition for alarm.

**Note**

For example, if sensitivity is 0, than frame on which fire/smoke is recognized with certainty more than 95% will be alarming. If sensitivity is 100, than frame with 50% certainty of fire/smoke recognition will be alarming.

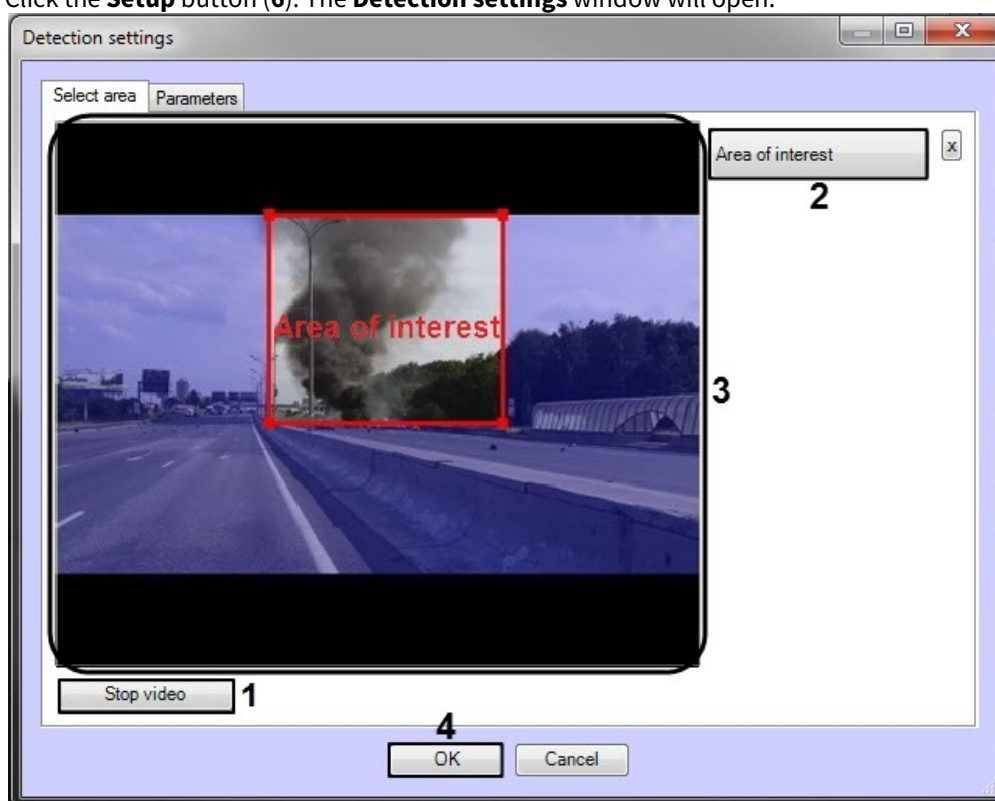
4. In the **Time in seconds between processed frames** field (3) enter the time period in seconds separating frames used by algorithm for fire/smoke analysis. This parameter has an impact on processor capacity: the less the parameter, the more capacity. Also it has an impact on operation speed of algorithm: the less distance between frames. the faster required number of frames will be collected for decision and alarm, if it's required.

5. In the **Number of frames for analysis and output** field (4) enter minimum number of frames for detection analysis before alarm generating. The more the value, the more certain the result of detection operation. At the same time, if value is too large, the short-time fire/smoke can be missed.

**Note.**

Multiplication of **Time in seconds between processed frames \* Number of frames for analysis and output** parameters characterizes time period (in seconds) at the end of which after fire/smoke alarm will be triggered.

6. In the **Device** drop-down list (5), select the device on which the fire/smoke detection will operate:
  - a. **Auto** - automatic device selection.
  - b. **CPU** - Central Processing Unit from Intel (not recommended).
  - c. **Nvidia GPU** (depends on the number of Nvidia graphics cards installed).
7. Specify the detection surveillance area on the video image:
  - a. Click the **Setup** button (6). The **Detection settings** window will open.



- b. Click the **Stop video** button to capture the video image (1).
- c. Click the **Area of interest** button (2).
- d. Specify area on which fire/smoke recognition will be detected (3).
- e. Click the **OK** button (4).

8. Click the **Apply** button (7).

Configuring the **Fire detection** and **Smoke detection** modules is complete.

## 5.14 Fluid level detection

### 5.14.1 Functionality of the Fluid level detection module

The **Fluid level detection** module operates together with the **IntLab-Carriages** license plates recognition module which is the part of the *Auto-Intellect* software. For more details on this module, refer to the *Auto-Intellect* software Administrator's Guide (the most relevant version of this document is available in the [AxxonSoft documentation repository](#)).

The **Fluid level detection** module provides the following functionality:

1. Determination of the fluid level in carriages passing in the field of view of the thermal camera.
2. Record data on the fluid level into the Event viewer database. When operating together with the **IntLab-Carriages** recognition module, the data on the fluid level is also recorded into the *Auto-Intellect* database.
3. Putting titles indicating the fluid level onto the video image in the Video Surveillance Monitor.
4. Putting the vertical line indicating the fluid level onto the video image in the Video Surveillance Monitor (using the script).
5. Transfer of data about the fluid level to the *Auto-Intellect* software for display in the **Vehicle Tracer** interface window and then search for the LP recognition events in the database by the fluid level.

**Note**

This functionality is disabled if the **Fluid level detection** module is operating independently, i.e. without the **IntLab-Carriages** recognition module.

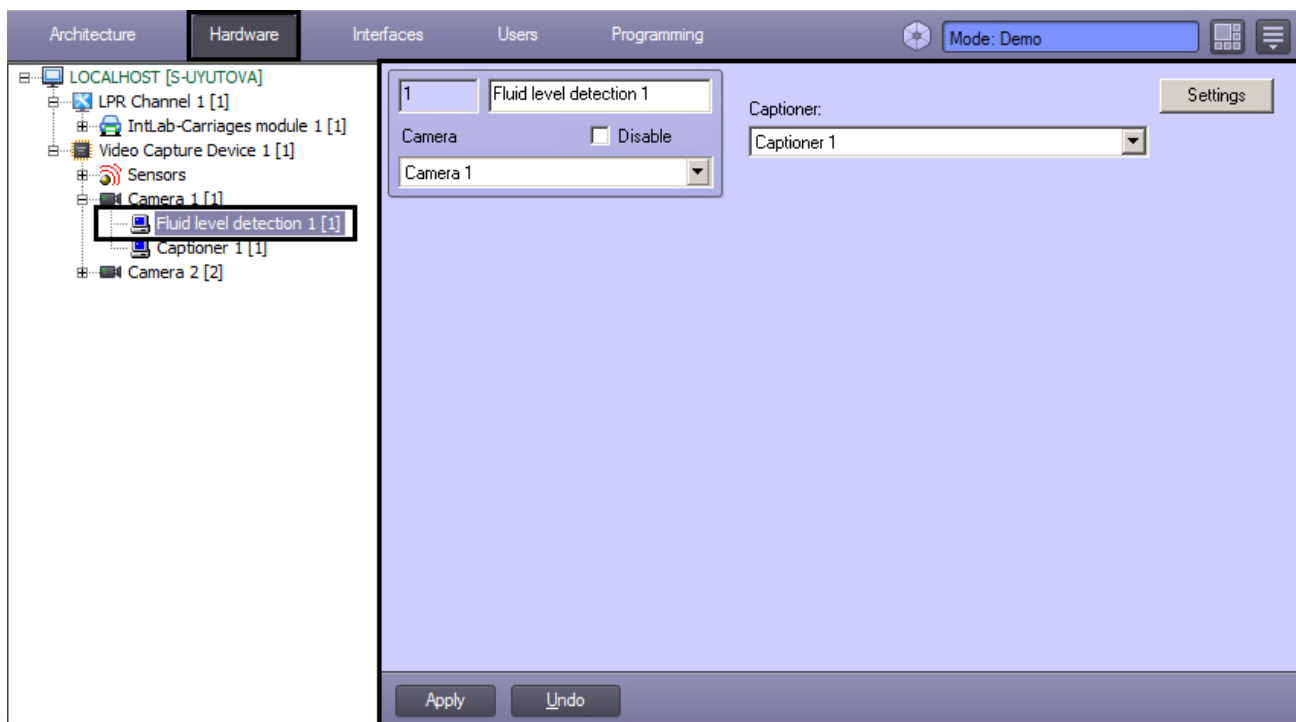
### 5.14.2 Camera requirements for the Fluid level detection module

The requirements for video cameras working with the **Fluid level detection** module are as follows:

Camera	<ul style="list-style-type: none"> <li>• Thermal imaging camera</li> </ul>
Scene and camera view	<ul style="list-style-type: none"> <li>• The optical axis of the video camera should be directed perpendicular to the direction of carriages movement</li> </ul>
Images of objects	<ul style="list-style-type: none"> <li>• Video should have a clear temperature gradient</li> </ul>

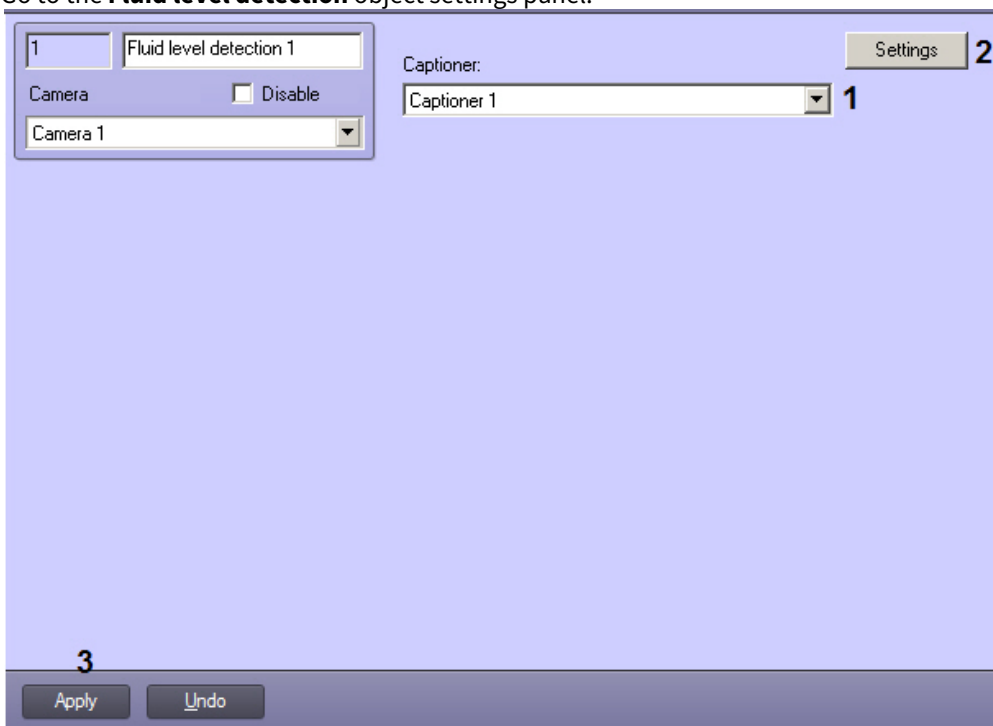
### 5.14.3 Configuring the Fluid level detection module

The *Fluid level detection* module is configured on the **Fluid level detection** object settings panel. This object is created under the **Camera** object on the **Hardware** tab of the **System settings** window.



The *Fluid level detection* module is configured as follows:

1. Go to the **Fluid level detection** object settings panel.

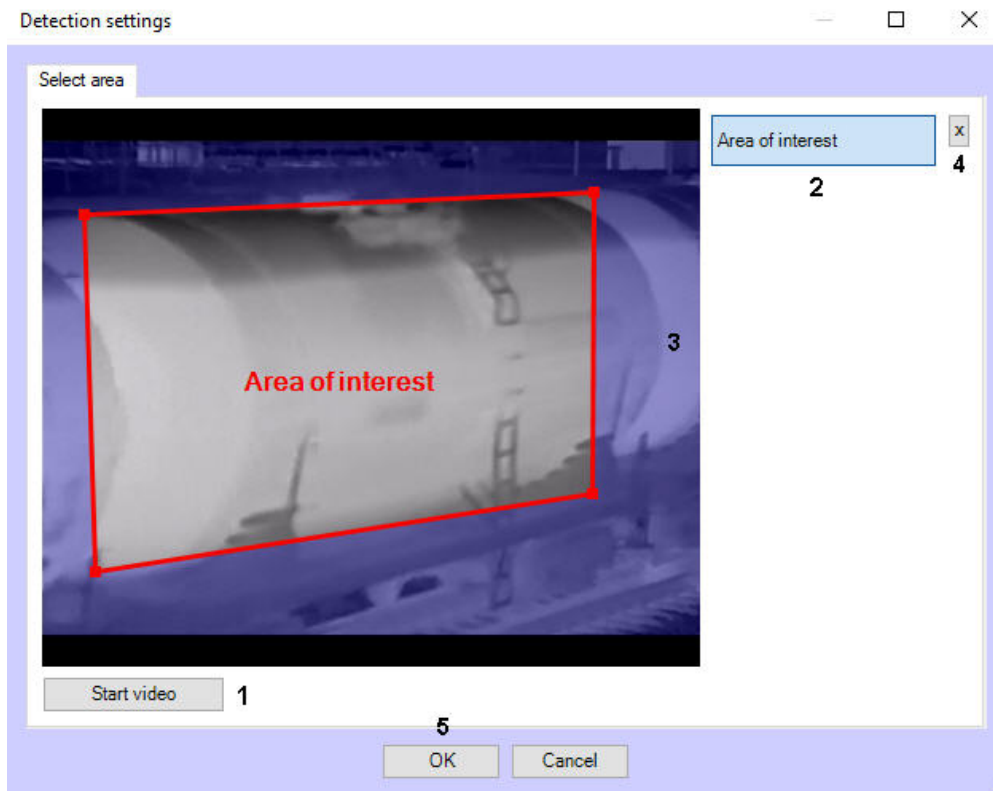


2. In the **Captioner** drop-down list select the **Captioner** object created under the same **Camera** object as the **Fluid level detection** object (1). This captioner will be used to overlay captions of fluid level onto the camera video image in the Video Surveillance Monitor.

**Note.**

For more info on how to create and configure the **Captioner** and **Monitor** objects refer to the *Intellect software. Administrator's Guide*. For details on operation of these objects refer to *Intellect software. Operator's Guide*. The most relevant versions of these documents are available in the [AxxonSoft documentation repository](#)).

- Click **Settings (2)**. The **Detection settings** dialog box opens.



- Select the frame in this settings window to set the area of interest. For that, click the **Start video** button (1).
- Click the **Area of interest** button (2).
- Using the left mouse button select the four corners of the area on the captured video image (3). Only one area may be so designated. Upon selection of the area the remaining part of the video image will be dimmed.

**Note.**

To remove a selected area, click the **x** button (4).

- Click **OK** (5).
- Click **Apply** on the **Fluid level detection** object settings panel to save changes.

After the fluid level detection tool is configured, it can be assigned to the **IntLab-Carriages** module for joint operation with the *Auto-Intellect* software. To do this, select the created and configured **Fluid level detection** object on the **IntLab-Carriages** module settings panel (see [Setting up the IntLab-Carriages module](#)).

## 5.15 Neurotracker

### 5.15.1 Functionality of the Neurotracker module

A neurotracker is designed for detecting any objects in a video image and calculating the metadata. The neurotracker operation involves the use of a neural network, which allows to reduce the computation load and improve the quality of the object detection. Any detection tool can be implemented on the basis of the neurotracker. The neurotracker independently receives the necessary tracks, both with moving and with completely motionless (during the whole analysis period) objects.

The following objects can be created on the basis of the neurotracker:

- Neurotracker counter (for periodical events notifying about the number objects received from the neurotracker).
- VMDA detection tools: Line crossing, Motion in the area (you can configure some actions in the system on triggering of VMDA detection tools).

## 5.15.2 Camera requirements for the Neurotracker module

The requirements for the cameras that will work with the **Neurotracker** module are listed in the following table.

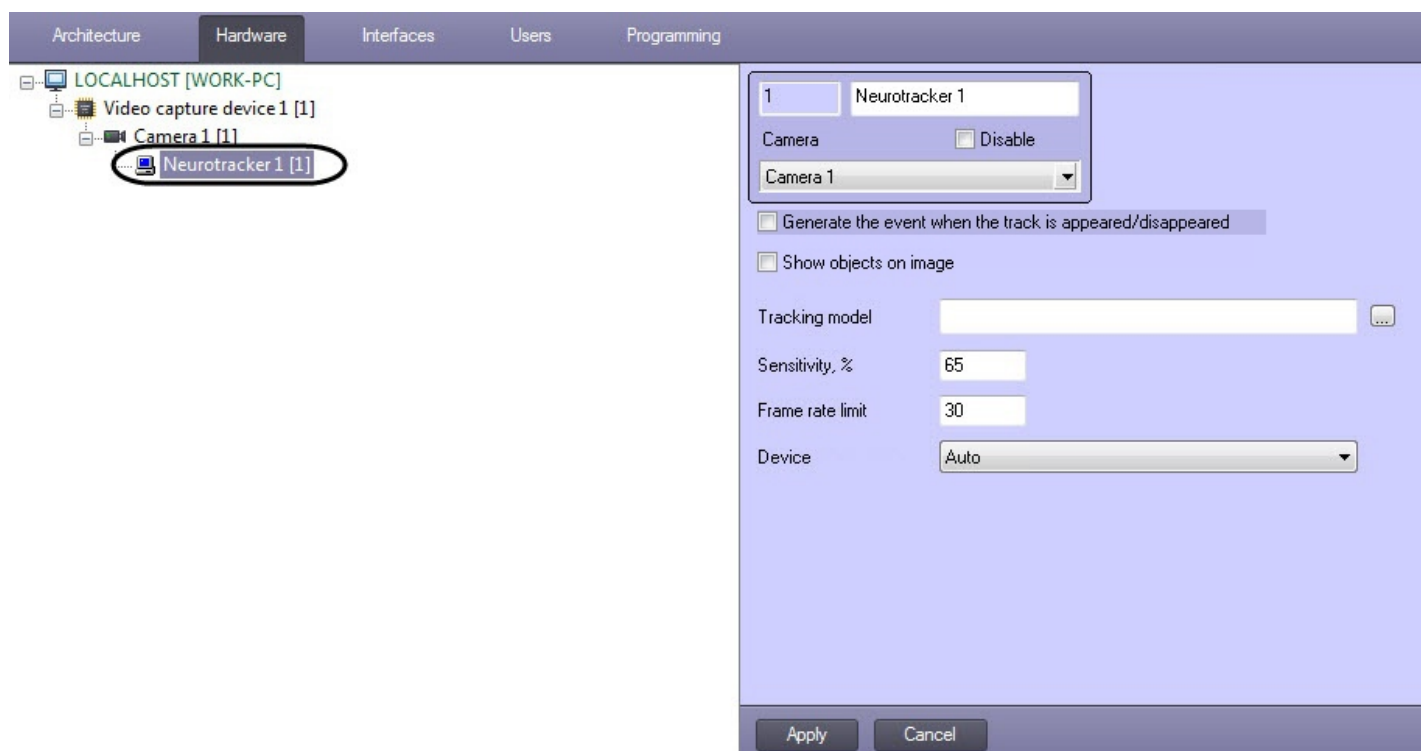
Camera	<ul style="list-style-type: none"> <li>Resolution: 640x480 pixels.</li> <li>Correct operation is not guaranteed with fish-eye lenses</li> <li>Fps: not less than 2. For detecting people, the recommended fps should be at least 6, for cars - at least 12.</li> </ul>
Lighting	<ul style="list-style-type: none"> <li>The objects must be visually separated from the background and from each other</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>The camera shall be installed at an angle at a height of at least 3 meters, or on the ceiling, or facing strictly down (for counting visitors)</li> </ul>
Objects image	<ul style="list-style-type: none"> <li>Minimal object size (the area of a rectangular track): at least 3% of the frame area.</li> </ul>

## 5.15.3 Configuring the Neurotracker module

### Attention!


The Neurotracker software module works only in *Intellect* 4.11.0 and higher version.

The **Neurotracker** software module is configured on the **Neurotracker** object settings panel. This object is created under the **Camera** object on the **Hardware** tab of the **System settings** window.



The **Neurotracker** software module is configured as follows:

1. Go to the **Neurotracker** object settings panel.

2. Set the **Generate the event when the track is appeared/disappeared** checkbox (1) if it is necessary to generate the event when the track appeared/disappeared.
3. Set the **Show objects on image** checkbox (2) if it is necessary to highlight the detected object with a frame in the **Monitor** interface object window.
4. Click the  button (3), and in the standard Windows window that opens, select the neural network file with the tracking model.

#### Notes

- If you intend to use a CPU or Intel GPU (integrated video core) for the neurotracker operation, then it is necessary to train the neural network file using the OpenVINO toolkit, and the name of the neural network file should contain the **\_openvino** substring at the end. For example, **test1\_openvino.ann**.
- If you intend to use the Nvidia GPU for the neurotracker operation, then the name of the neural network file should contain only the name of the neural network. For example, **test1.ann**.
- If you intend to use the Intel Neural Compute Stick (Movidius of any generation) or Intel HDDL (a device for working with several Movidius) for the neurotracker operation, then the name of the neural network file should contain the **\_movidius** substring at the end. For example, **test1\_movidius.ann**.

#### Attention!

Also, there should be a file with the **\*.txt** extension in the same directory as the neural network file with the **\*.ann** extension. The **\*.txt** file should have the same name as the **\*.ann** file.

5. In the **Sensitivity, %** field (4), specify the neurotracker sensitivity — an integer value within the range from **0** to **100**.

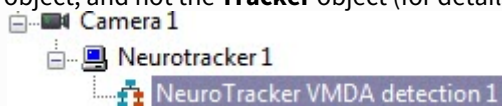
**Note**

The neurotracker sensitivity is determined experimentally. The lower the sensitivity, the more false triggerings there might be. The higher the sensitivity, the less false triggerings there might be, however, some useful tracks might be skipped.

6. In the **Frame rate limit** field (5), specify the number of frames per second to be analyzed by the neural network. All other frames will be interpolated. The higher the specified value, the more accurate the tracking, but the higher the CPU load.
7. In the **Device** drop-down list (6) select the device where the neurotracker will operate:
  - **Auto** — automatic device selection.
  - **CPU** — Intel central processing unit (not recommended).
  - **Nvidia GPU** (depends on the number of installed Nvidia video cards).
  - **Movidius** — Intel Neural Compute Stick or Intel HDDL.
  - **IntelGPU** — the embedded video core of the Intel CPU.
8. Click the **Apply** button (7).

**Note**

- If necessary, create and configure the NeuroTracker VMDA detection tools on the basis of the **Neurotracker** object. The procedure of creating and configuring the NeuroTracker VMDA detection tools is similar to creating and configuring the VMDA detection tools for a regular tracker. The only difference is that it is necessary to create the NeuroTracker VMDA detection tools on the basis of the **Neurotracker** object, and not the **Tracker** object (for details, see [Creating and configuring VMDA detection](#)).



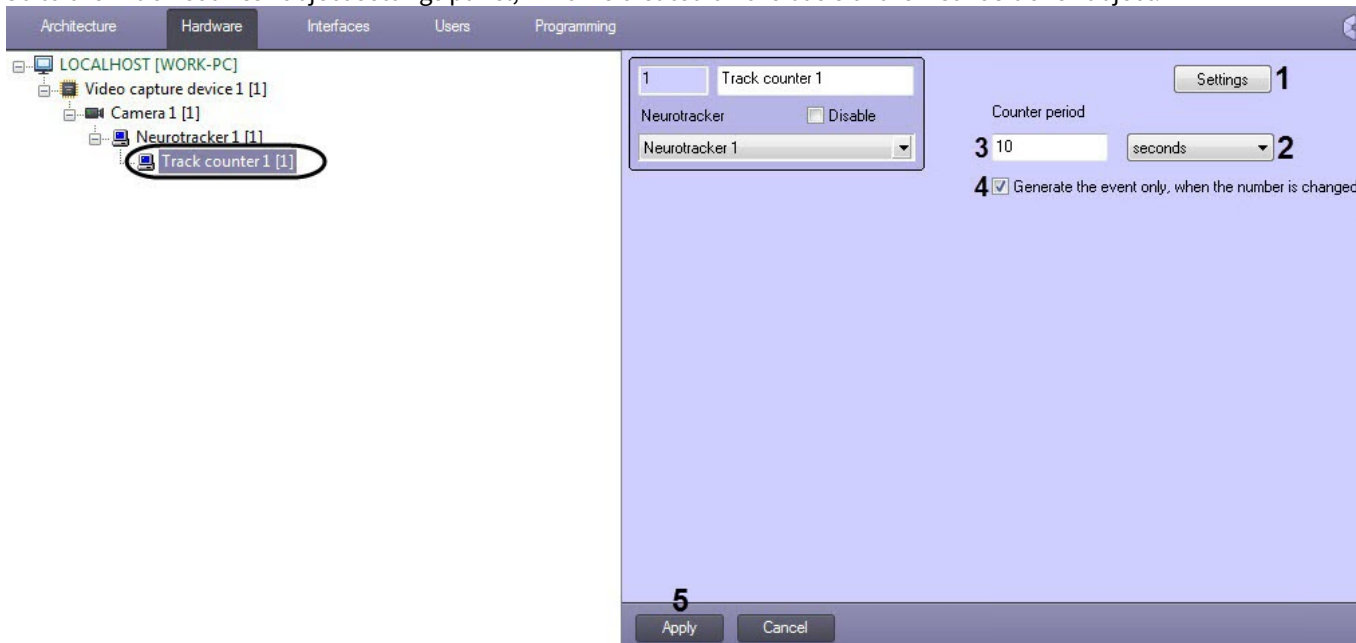
- If it is necessary to periodically generate the events containing the number of objects received from the neurotracker, then create and configure the neurotracker track counters (see [Configuring the neurotracker track counter](#)).

The **Neurotracker** software module configuration is complete.

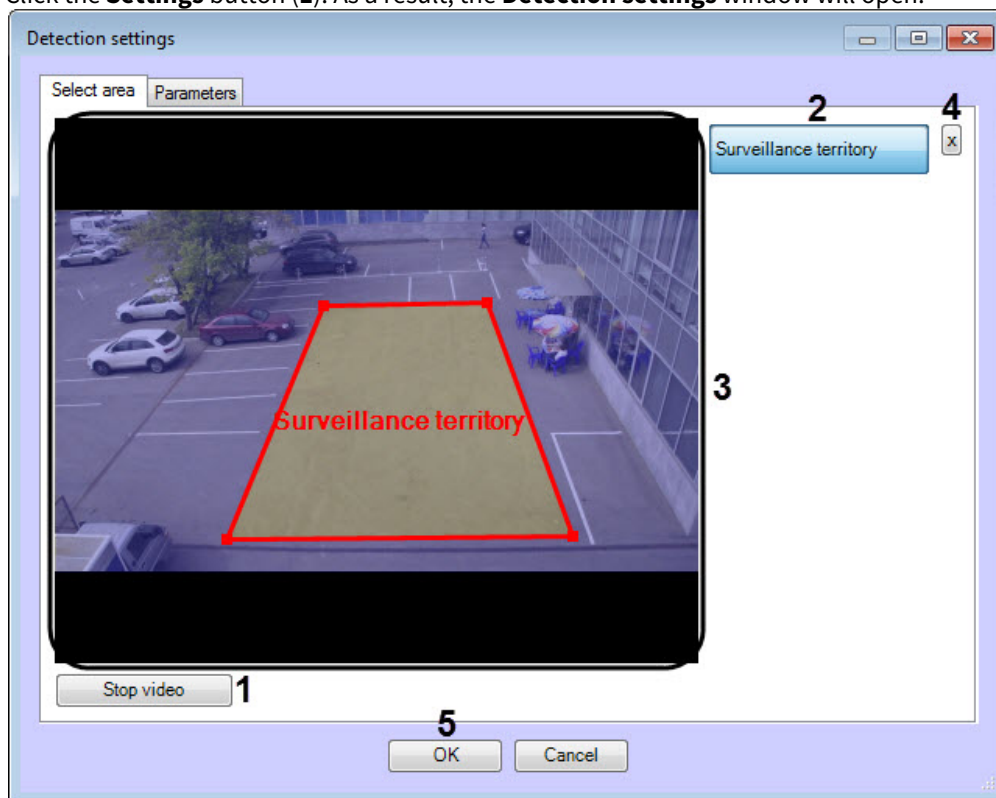
### 5.15.3.1 Configuring the neurotracker track counter

The neurotracker track counter is configured as follows:

1. Go to the **Track counter** object settings panel, which is created on the basis of the **Neurotracker** object.




2. On the video image, specify the surveillance area, within which the track counter will count the objects:
  - a. Click the **Settings** button (1). As a result, the **Detection settings** window will open.



- b. Click the **Stop video** button (1) to capture a frame of the video image.
- c. Click the **Surveillance territory** button (2) and select the area, within which the track counter will count the objects (3).

**Note**

To delete the selected area, click the  button (4).

- d. Click **OK** button (5).
3. From the drop-down list (2), select the counter period time unit: **seconds, minutes, hours, days**.
4. In the **Counter period** field (3), specify the time in which the event containing the total number of objects recorded for a given time period will be generated.
5. Set the **Generate the event only, when the number is changed** checkbox (4) to generate the events only if the number of recorded objects has changed.

**Note**

If this check box is set, the **Counter period** field becomes inactive, and the specified value of the counter period is ignored.

6. Click the **Apply** button (5).

The neurotracker track counter is configured.

## 5.16 Person location tracker

### 5.16.1 Functionality of the Person location tracker module

Person location tracker — the detection module for tracking the human location on the video image using the neural network which allows reducing the processing load and improving the detection quality. The Person location tracker allows detecting the human pose and generating the corresponding event.

On the basis of the Person location tracker, the **Pose detection** objects can be created including the following detections:

- Sitting person detection — the detection triggers when a sitting human is found in the frame.
- Recumbent person detection — the detection triggers when a recumbent human is found in the frame.
- Hands up detection — the detection triggers when a human with one or two hands raised is found in the frame. The hand is considered to be raised if the forearm is parallel to the backbone.
- Active shooter detection — the detection triggers when a human reaching his/her hand forward parallel to the ground is found in the frame.
- Person detection — the detection triggers when any pose of a human is found in the frame.
- Close-standing people detection — the detection triggers if the distance between two people is less than specified.

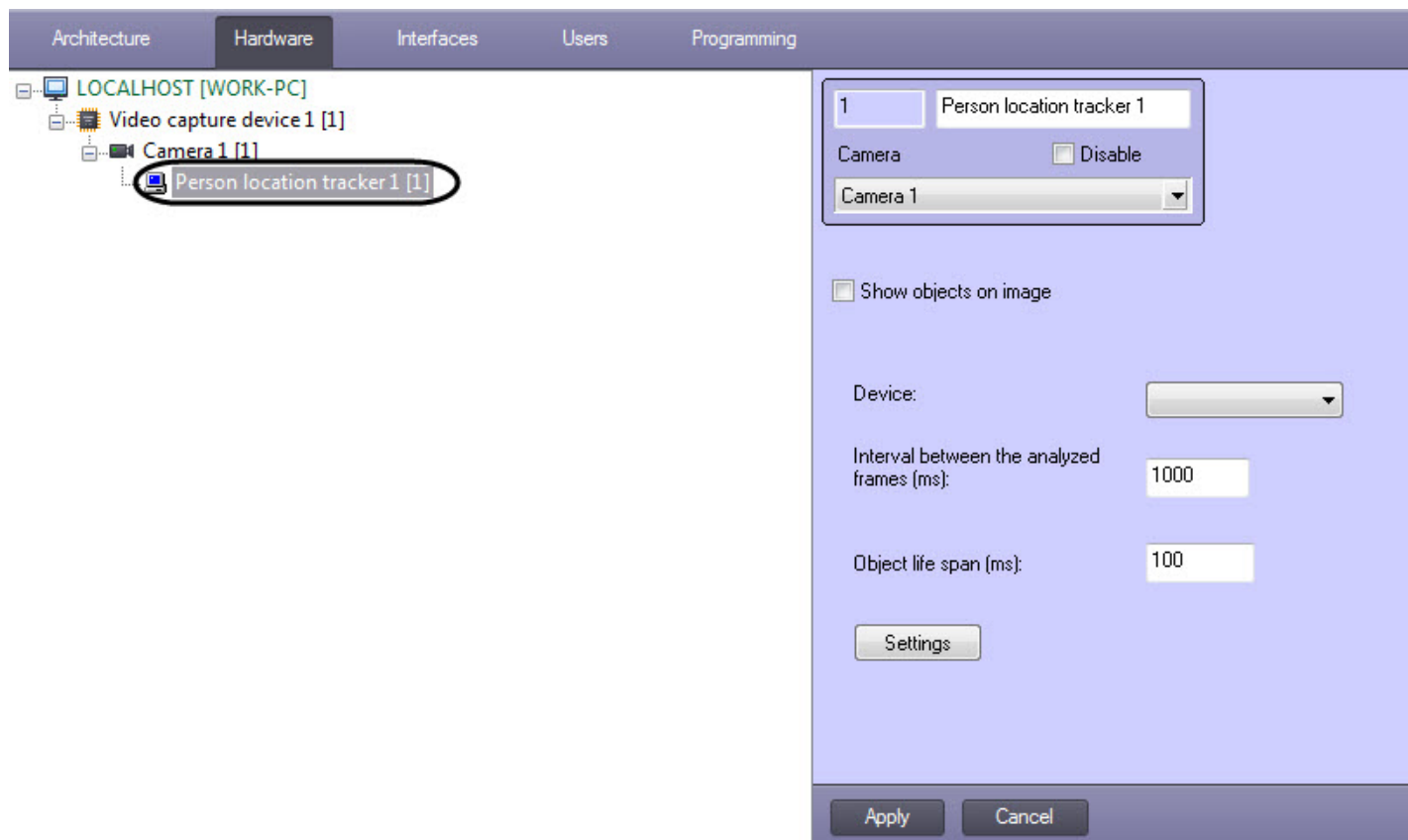
### 5.16.2 Camera requirements for the Person location tracker module

The requirements for the cameras that will work with the **Person location tracker** module are listed in the following table.

Camera	<ul style="list-style-type: none"> <li>• Resolution: 640x480 pixels.</li> <li>• Correct operation is not guaranteed with fish-eye lenses</li> <li>• Fps: not less than 2 (recommended: 4 or more)</li> </ul>
Lighting	<ul style="list-style-type: none"> <li>• The objects must be visually separated from the background and from each other</li> </ul>
Scene and camera angle	<ul style="list-style-type: none"> <li>• The camera shall be installed at an angle at a height of at least 3 meters, or on the ceiling, or facing strictly down (for counting visitors)</li> </ul>
Objects image	<ul style="list-style-type: none"> <li>• Minimal object size (the area of a rectangular track): at least 10% of the frame area.</li> </ul>

### 5.16.3 Configuring the Person location tracker module

The **Person location tracker** software module is configured on the **Person location tracker** object settings panel. This object is created under the **Camera** object on the **Hardware** tab of the **System settings** window.

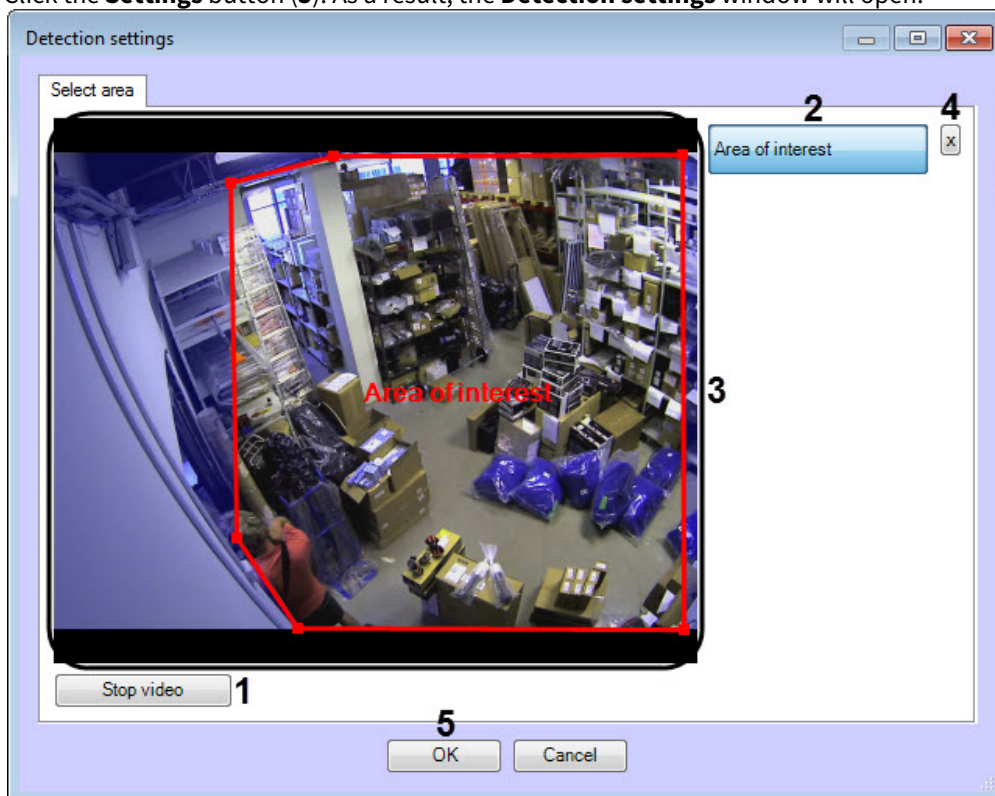


The **Person location tracker** software module is configured as follows:

1. Go to the **Person location tracker** object settings panel.


2. Set the **Show objects on image** checkbox (1) if it is necessary to highlight the detected object with a frame in the **Monitor** object interface window.
3. In the **Device** drop-down list (2) select the device where the person location tracker will operate:
  - **Auto** — automatic device selection.
  - **CPU** — central processing unit (not recommended).
  - **Nvidia GPU** (depends on the number of installed Nvidia video cards).
  - **Movidius** - Intel Neural Compute Stick or Intel HDDL.
  - **IntelGPU** - the embedded video core of the Intel CPU.
4. In the **Interval between the analyzed frames (ms)** field (3), enter the interval between the analyzed frames in milliseconds. The value must be in the range of 30 to 10000.
5. In the **Object life span (ms)** field (4), enter the time in milliseconds in which the found object will be considered as disappeared from the frame.

6. Click the **Settings** button (5). As a result, the **Detection settings** window will open.



- a. Click the **Stop video** button (1) to capture the video image.
- b. Click the **Area of interest** button (2) and specify the area in which the Person location tracker will operate (3).

**Note**

To delete the selected area, click the  button (4).

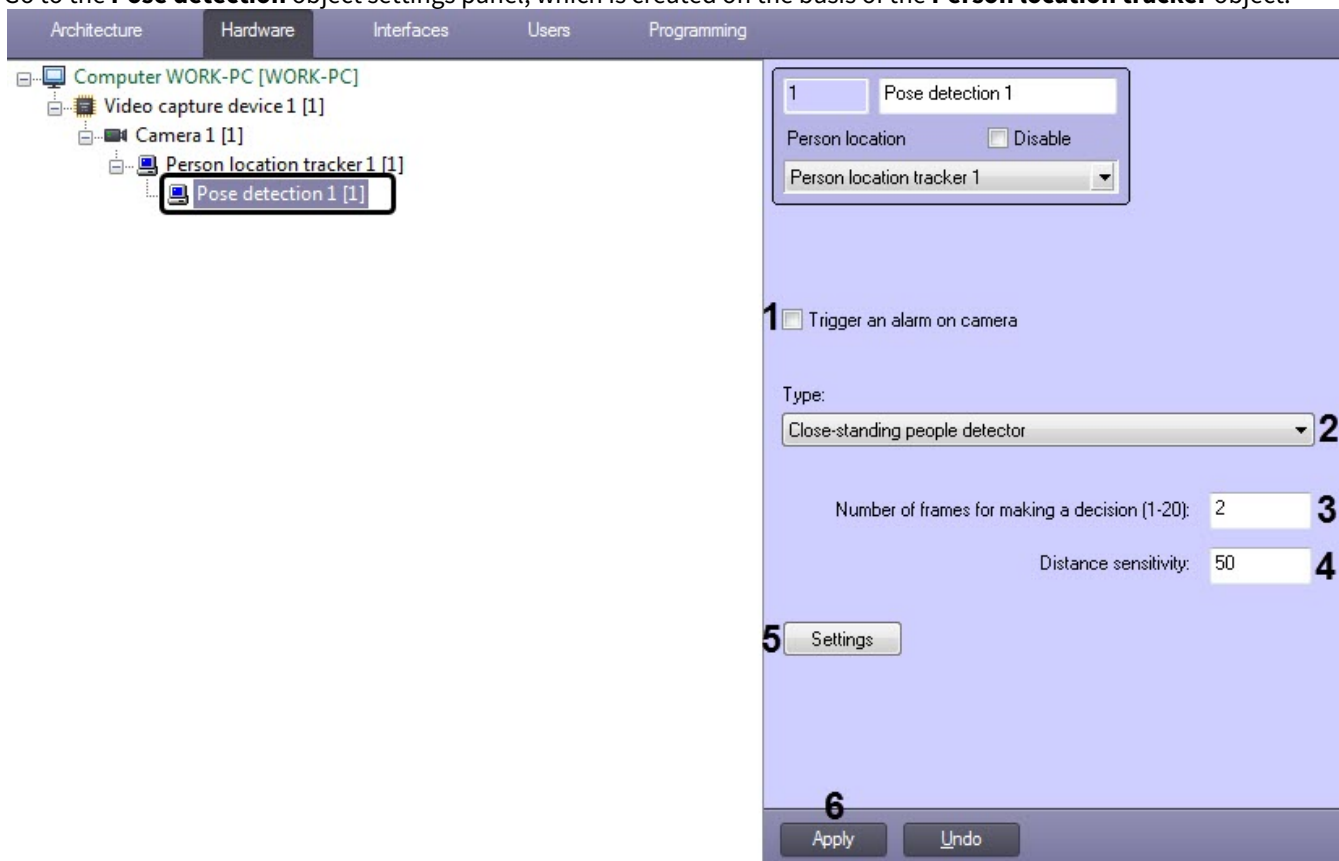
- c. Click **OK** button (5).

7. Click the **Apply** button (6).
8. Perform pose detection configuration (see [Configuring the pose detection](#)).

### 5.16.3.1 Configuring the pose detection

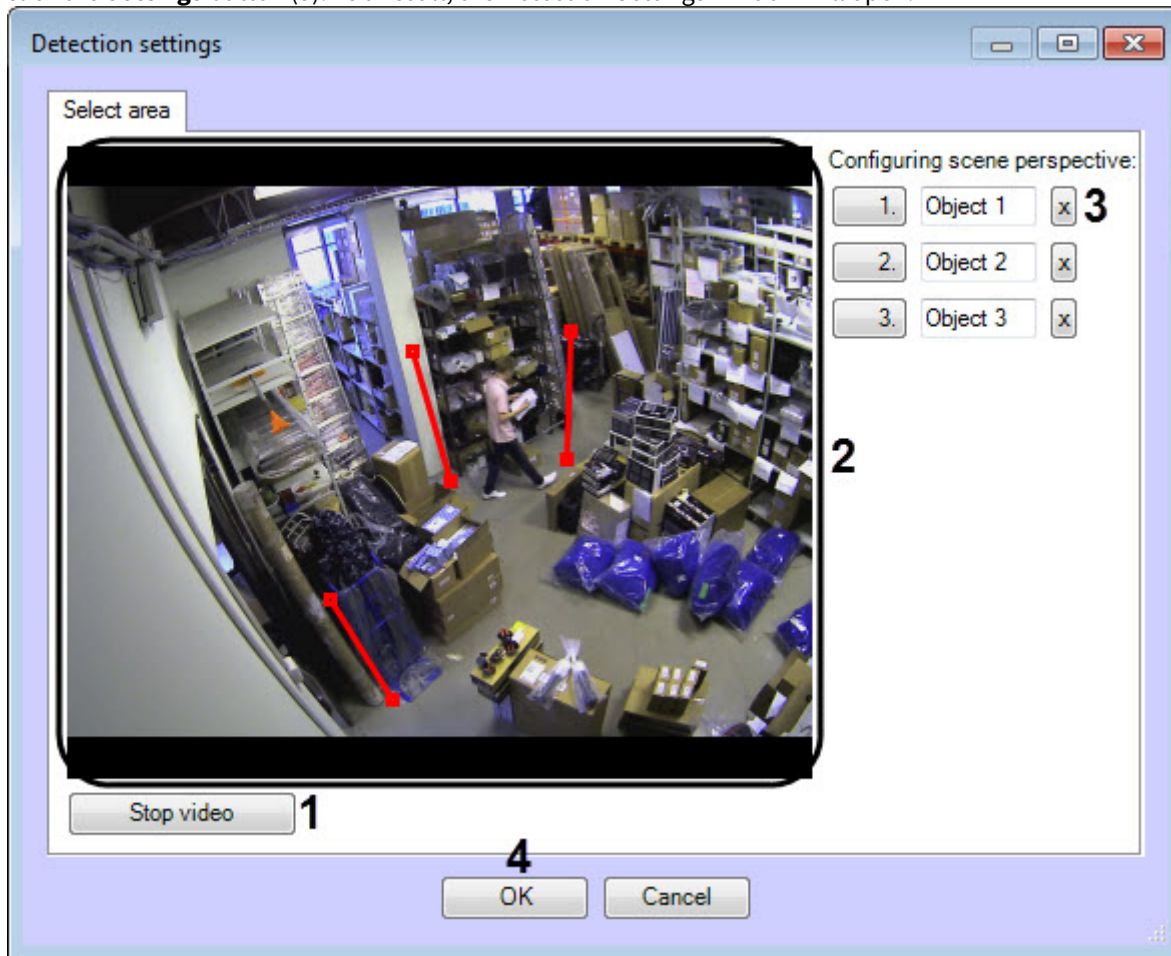
The pose detection is configured as follows:

1. Go to the **Pose detection** object settings panel, which is created on the basis of the **Person location tracker** object.




2. Set the **Trigger an alarm on camera** check box (1) if it is also necessary to trigger an alarm on the camera by the specified pose detection.
3. From the **Type** drop-down list (2), select the human type which is necessary to detect:
  - **Sitting person detection** — the detection triggers when a sitting human is found in the frame.
  - **Recumbent person detection** — the detection triggers when a recumbent human is found in the frame.
  - **Hands up detection** — the detection triggers when a human with one or two hands raised is found in the frame. The hand is considered to be raised if the forearm is parallel to the backbone.
  - **Active shooter detection** — the detection triggers when a human reaching his/her hand forward parallel to the ground is found in the frame.
  - **Person detection** — the detection triggers when any pose of a human is found in the frame.
  - **Close-standing people detection** — the detection triggers if the distance between two people is less than specified.
4. In the **Number of frames for making a decision (1-20)** field (3), enter the required amount of frames from 1 to 20 on which the human must be found in the specified pose to trigger the corresponding event.
5. In the **Distance sensitivity** field (4), enter the distance sensitivity value between people in the range from 0 to 500. It is used only if the **Close-standing people detection** is selected.
6. Specify the perspective on the video image on which the human location will be detected:

- a. Click the **Settings** button (5). As a result, the Detection Settings window will open.



- b. Click the **Stop video** button (1) to capture the video image.
- c. Specify the size of the same human at least in three different parts of the frame (2) with the calibration lengths. To create a calibration length, it is necessary to click and hold the left mouse button on the video image so that you can specify the necessary length size. Changing the size of the already created lengths is executed by using nodes.

**Note**

To delete the created length, it is necessary to click the  button (3).

- d. Click the **OK** button (4).

7. Click the **Apply** button (6).

The pose detection is configured.

## 6 Operating with detection modules

### Note.

In order to view alarm archive or frame for events of detection modules supporting events in *Event Viewer* or *Operator protocol*, create and configure the detection tool object link with camera – see *Intellect software. Administrator's Guide*, the most relevant version of this document is available in the [AxxonSoft documentation repository](#).

### 6.1 Operating the «Queue length detection» module

#### 6.1.1 Obtaining traffic information in the area of interest

Events from the **Queue length detection** module get at specified intervals to the **Event log**.

Source	Event	Partition	Add. info	Date	Time
Camera 1	Alarm	Region 1		12-05-12	10:22:46
Queue Length Detection 1	Queue full		0	12-05-12	10:23:17
Queue Length Detection 1	Queue full		0	12-05-12	10:23:27
Queue Length Detection 1	Queue full		5	12-05-12	10:23:37
Queue Length Detection 1	Queue full		15	12-05-12	10:23:47
Queue Length Detection 1	Queue full		19	12-05-12	10:23:57
Queue Length Detection 1	Queue full		22	12-05-12	10:24:07
Queue Length Detection 1	Queue full		24	12-05-12	10:24:17
Queue Length Detection 1	Queue full		25	12-05-12	10:24:27
Queue Length Detection 1	Queue full		25	12-05-12	10:24:37
Queue Length Detection 1	Queue full		24	12-05-12	10:24:47
Queue Length Detection 1	Queue full		24	12-05-12	10:24:57
Queue Length Detection 1	Queue full		26	12-05-12	10:25:07
Queue Length Detection 1	Queue full		25	12-05-12	10:25:17

### Note

For more information on working with the **Event log** interface, see the [Operator's Guide](#).

#### 6.1.2 Generating a report on the traffic in the area of interest

Reports on the traffic in the area of interest are generated via the web-based *Report System*.

All necessary information is provided in the web-based [Intellect Web Report System. User Guide](#).

#### 6.1.3 Visualization of operating the Queue length detection

Visualization of operating the Queue length detection in the Monitor window can be realized with the help of user scripts on the base of **Titles** object. Detailed description of one of these scripts is presented in the [1.7 Examples of scripts on the Jscript language section of Programming guide \(Jscript\) document – Example 1. Visualization of operating the Queue length detection in the Video surveillance monitor](#).



## 6.2 Operating the «People counter detection» module

### 6.2.1 Obtaining information on number of visitors

The **People counter detection** module provides entries onto the **Event log** when visitors pass through the area of interest.

Source	Event	Partition	Add. info	Date	Time
Camera 1	Alarm	Region 1		12-05-12	10:29:35
People Counter Detection 1	Visitor exit			12-05-12	10:30:40
People Counter Detection 1	Visitor exit			12-05-12	10:30:43
People Counter Detection 1	Visitor exit			12-05-12	10:30:52
People Counter Detection 1	Visitor entrance			12-05-12	10:31:19
People Counter Detection 1	Visitor exit			12-05-12	10:31:42
People Counter Detection 1	Visitor entrance			12-05-12	10:31:43
People Counter Detection 1	Visitor exit			12-05-12	10:32:03
People Counter Detection 1	Visitor exit			12-05-12	10:32:15
People Counter Detection 1	Visitor entrance			12-05-12	10:32:32

When a visitor moves from sector 1 to sector 2, it is logged as **Visitor entry**; if the visitor moves from sector 2 to sector 1, it is logged as **Visitor exit**.

**Note.**

For more information on working with the **Event log** interface, see [Operator's Guide](#).

### 6.2.2 Generating a visitor report

Visitor reports are generated via the web-based *Report System*.

All necessary information is provided in the web-based [Intellect Web Report System. User Guide](#).

### 6.2.3 Visualization of operating the People counter detection

Visualization of operating the People counter detection in the Monitor window can be realized with the help of user scripts on the base of **Titles** object. Detailed description of one of these scripts is presented in the [1.7 Examples of scripts on the Jscript language section of Programming guide \(Jscript\) document – Example 2. Visualization of operating the People counter detection in the Video surveillance monitor.](#)



### 6.3 Operating the «Stopped vehicle detection» module

The **Stopped vehicle detection** module sends messages to the **Event log** when the stopped vehicle is detected in the surveillance area.

Source	Event	Partition	Add. info	Date	Time
● Stopped cars detector 1	Stopped car			27-11-12	13:18:11
● Stopped cars detector 1	Stopped car			27-11-12	13:18:15
Stopped cars detector 1	Traffic jam			27-11-12	13:18:20
Stopped cars detector 1	Traffic jam			27-11-12	13:18:21
● Stopped cars detector 1	Stopped car			27-11-12	13:18:29
● Stopped cars detector 1	Stopped car			27-11-12	13:18:32
● Stopped cars detector 1	Stopped car			27-11-12	13:18:36
Stopped cars detector 1	Traffic jam			27-11-12	13:18:37
Stopped cars detector 1	Traffic jam			27-11-12	13:18:38
Stopped cars detector 1	Traffic jam			27-11-12	13:18:41

**Note.**

For more information on working with the **Event log** interface, see the [Operator's Guide](#).

When a traffic jam is detected, it is logged as a **Traffic jam** event.

If a stopped car is detected in the surveillance area, it is logged as a **Stopped car** event.

**Note.**

The independent `_zone<>` parameter of the STOPPEDCAR (**Stotted car**) event contains the id number of a zone where the stopped car was detected. This parameter can be used in macros, programs and scripts. More details on these tools are available in the *Intellect Software. Administrator's Guide*, *Intellect Software. Programming Guide* and *Intellect Software. Programming Guide (JScript)*. The most recent versions of these documents are available at [AxxonSoft documentation repository](#).

## 6.4 Operating the «Glow detection» module

The **Glow detection** module sends messages to the **Event viewer** when the light source is detected or lost in the surveillance area.

Source	Event	Partition	Add. info	Date	Time
Glow detection 1	Disabled		3	07-05-13	10:14:55
Glow detection 1	Enabled		1	07-05-13	10:14:56
Glow detection 1	Disabled		2	07-05-13	10:14:56
Glow detection 1	Disabled		3	07-05-13	10:14:56
Glow detection 1	Disabled		1	07-05-13	10:14:57
Glow detection 1	Disabled		2	07-05-13	10:14:57
Glow detection 1	Disabled		3	07-05-13	10:14:57
Glow detection 1	Enabled		2	07-05-13	10:14:59

When a light source is detected (enabled), it is logged as an **Enabled** event. If a light source is lost (disabled), it is logged as a **Disabled** event. The number of surveillance area from which the event received is displayed in the **Add. Info** column.

## 6.5 Operating the «Heat map detection»

### 6.5.1 Generating a report on the basis of Heat map detection

Reports are generated via the web-based *Report System*.

All necessary information is provided in the web-based [Intellect Web Report System. User Guide](#).

## 6.6 Operating the «Detection of moving against crowd flow»

The **Detection of moving against crowd flow** module sends the **Moving against crowd flow** message to the **Event viewer** when the objects moved against a crowd are recognized in the monitored area.

Source	Event	Region	Add. info	Date	Time
Detection of moving against crowd flow 1	Moving against crowd flow		0	01-10-15	16:27:52
Detection of moving against crowd flow 1	Moving against crowd flow		0	01-10-15	16:28:07

**Note.**

For more information on working with the **Event viewer** interface, see the [Operator's Guide](#).

## 6.7 Operating the «Sweethearting detection» module

In case the scanned item appears in the area of interest, the **Item at checkout** message is sent from the **Sweethearting at checkout detection** module to the **Event viewer** interface window.

Event viewer 1						<input type="checkbox"/> Show filters	Clear
Source	Event	Region	Add. info	Card	Date and time		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.998158}]}		25.09.2018 14:50:02		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.999652}]}		25.09.2018 14:50:05		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.999421}]}		25.09.2018 14:50:08		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.998898}]}		25.09.2018 14:50:11		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.963029}]}		25.09.2018 14:50:14		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.998641}]}		25.09.2018 14:50:17		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.985236}]}		25.09.2018 14:50:20		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.997084}]}		25.09.2018 14:50:23		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.999842}]}		25.09.2018 14:50:26		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.998111}]}		25.09.2018 14:50:29		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.999847}]}		25.09.2018 14:50:32		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.994226}]}		25.09.2018 14:50:35		
Sweethearting at checkout detection 1	Item at checkout		{"status": "ok", "detectedObjects": [{"label": "product", "score": 0.983732}]}		25.09.2018 14:50:38		

The detailed information about working with **Event viewer** interface window is presented in the *Intellect software package. Operator's Guide*. Current version of this document is available in the [AxxonSoft documentation repository](#).

The **Sweethearting detection** module provides a timeout during which the repeated triggering of the detection tool is ignored. By default, the timeout duration is 3000 milliseconds. To change the timeout duration, specify the necessary value in milliseconds for the **report\_frequency** parameter (for more details, see [Registry keys reference guide](#), for more information about working with the registry, see [Working with Windows OS registry](#)).

### Note

If the timeout duration was changed, to apply the specified parameter value, it is necessary to restart the *Intellect* or click the **Apply** button on the **Sweethearting detection** module settings panel (see [Configuring the «Sweethearting detection» module](#)).

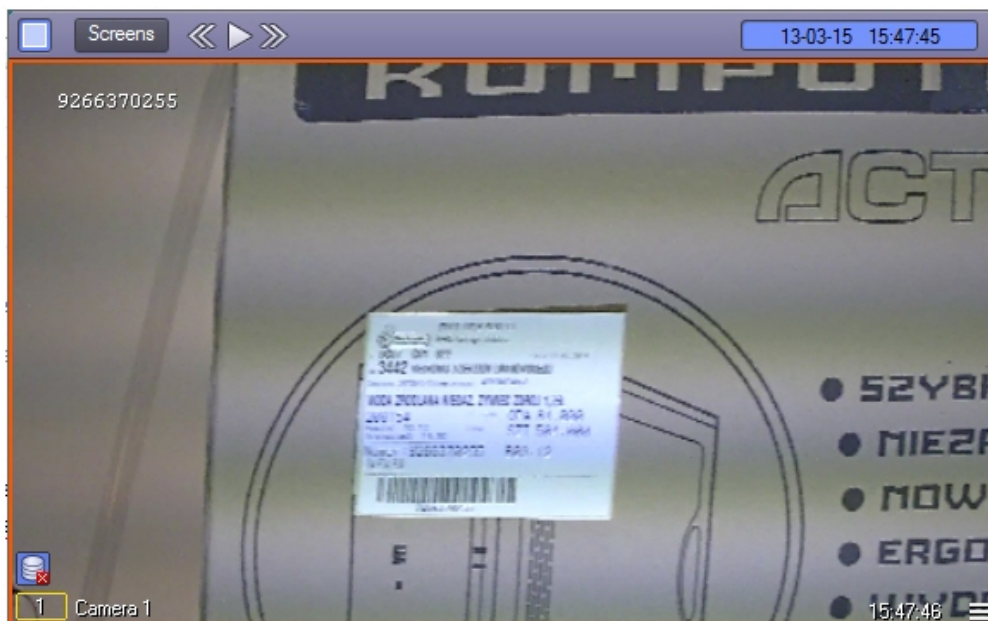
### 6.7.1 Generating sweethearting reports

Sweethearting reports are generated via the *Intellect Web Report System* (the Sweethearting report). For more details on creating reports refer to [Intellect Web Report System. User Guide](#).

In order to be able to create a Sweethearting report, **POS-terminal** objects corresponding to the cash terminals on which the cashiers operate should be configured. This object is part of the *POS-Intellect* software package (see [POS-Intellect. Administrator's Guide](#) for details, the most current version of this document is available in the [AxxonSoft documentation repository](#)).

## 6.8 Operating the «Barcode detection» module

In case of recognizing barcode or QR-code in the area of interest the result of the **Barcode detection** operation will be displayed in the **Monitor** interface objects using the **Captioner** object.



Search by detected barcodes is performed using the **Search by captions** interface object.

### 6.9 Operating the «Train detection» module

The **Train detection** module sends messages to the **Event viewer** interface window in case of train recognizing in the monitored area or in case of train disappearance from the monitored area.

Source	Event	Region	Add. info	Date	Time
Train detection 1	Train arrived			19-10-15	12:44:53
Train detection 1	Train departed			19-10-15	12:45:51
Train detection 1	Train arrived			19-10-15	12:47:38
Train detection 1	Train departed			19-10-15	12:48:37
Train detection 1	Train arrived			19-10-15	12:50:24
Train detection 1	Train departed			19-10-15	12:51:23

When the train is recognized, it is logged as an **Train arrived** event. When the train is disappeared, it is logged as an **Train departed** event.

### 6.10 Operating the "Fire detection" and "Smoke detection" modules

The **Fire detection/Smoke detection** module sends **Fire detected/Smoke detected** messages to the **Event viewer** interface window in case of fire/smoke recognizing in the monitored area. When fire/smoke disappears in the monitored area, the **Fire stopped/Smoke stopped** event is displayed. If fire/smoke is permanently present in the monitored area, only event of fire/smoke detection is displayed. If the detection tool is configured correctly, no other events from it comes until fire/smoke disappearance. See also [Configuring the «Fire detection» and «Smoke detection» modules](#).

Event viewer 1						<input type="checkbox"/> Show filters	Clear
Source	Event	Region	Add. info	Card	Date and time		
● Fire detection 1	Fire detected				04.10.2017 13:00:52		
● Fire detection 1	Fire stopped				04.10.2017 13:01:00		

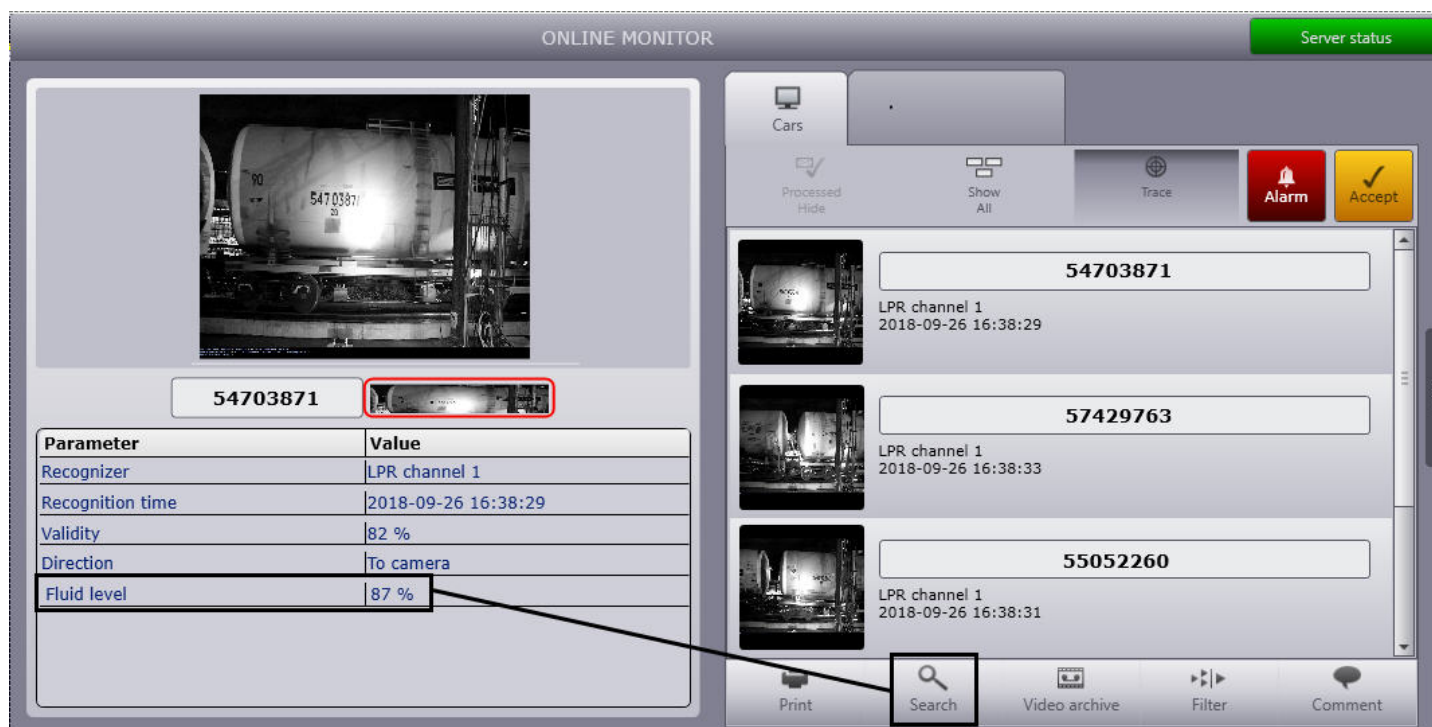
  

Event viewer 1						<input type="checkbox"/> Show filters	Clear
Source	Event	Region	Add. info	Card	Date and time		
● Smoke detection	Smoke detected				04.10.2017 13:00:52		
● Smoke detection	Smoke stopped				04.10.2017 13:01:00		

Detailed information about working with *Event viewer* interface window is presented in the [Operator's Guide](#). Current version of this document is available in the [AxxonSoft documentation repository](#).

### 6.11 Operating the «Fluid level detection» module

In case the **Fluid level detection** object is selected on the **IntLab-Carriages** module settings panel, then the *Fluid level detection* module is operated in the **Vehicle tracer** interface window which is the part of the *Auto-Intellect* software. For details on operation of this object refer to the *Auto-Intellect. Operator's Guide* (the most relevant versions of this document is available in the [AxxonSoft documentation repository](#)). The fluid level is displayed in the parameters list in the **Online monitor** window when the LP is being recognized. You can also perform a search by the fluid level in the recognizers databases.



The *Fluid level detection* module also displays the fluid level as captions over the video image in the Surveillance Window of the the corresponding camera.

**Note**

To use this functionality, specify the captioner when configuring the *Fluid level detection* module (see [Configuring the «Fluid level detection» module](#)).

In addition, a script can be created that allows drawing a vertical line on the video image in the Surveillance Window, showing the level of the fill:



For details on the scripts refer to the [Intellect software. Programming Guide \(JScript\)](#).

The script text is given below. After copying, replace the cam (camera identifier) and x1, x2, y1, y2 parameters with the actual values.

```

if(Event.SourceType == "FLUID_DETECTOR" && Event.SourceId == "1" && Event.Action ==
"FLUID_ACTION")
{
  var L = Event.GetParam("param0");
  L=100-L*100; //the L parameter sets the line height

  DoReactStr("MONITOR", "", "SET_MARKRECT", "cam<"+cam+">,color<255>,id<"+cam+">,x1<"+x1+">,x2<"+x2+">
>,y1<"+L+">,y2<"+y2+">");
  // coordinates x1, x2, y1, y2 set line position in the Surveillance window
}

```

## 6.12 Operating the «Neurotracker» module

In case the alarm is detected in the monitored area, the **Neurotracker** software module sends the **Alarm** message to the **Event viewer** interface window. When the alarm disappears in the monitored area, the **No alarm** event is displayed. See also [Configuring the «Neurotracker» module](#).

For details on operation of working with the **Event viewer** interface window, refer to the *Intellect Operator's Guide*. Current version of this document is available in the [AxxonSoft documentation repository](#).

## 6.13 Operating the Person location tracker module

In case the alarm is detected in the monitored area, the **Person location tracker** module sends the **Alarm** message to the **Event viewer** interface window. When the alarm disappears in the monitored area, the **No alarm** event is displayed. See also [Configuring the Person location tracker module](#).


For details on operation of working with the **Event viewer** interface window, refer to the *Intellect Operator's Guide*. Current version of this document is available in the [AxxonSoft documentation repository](#).

## 7 Appendix 1. Debug window

### 7.1 General information

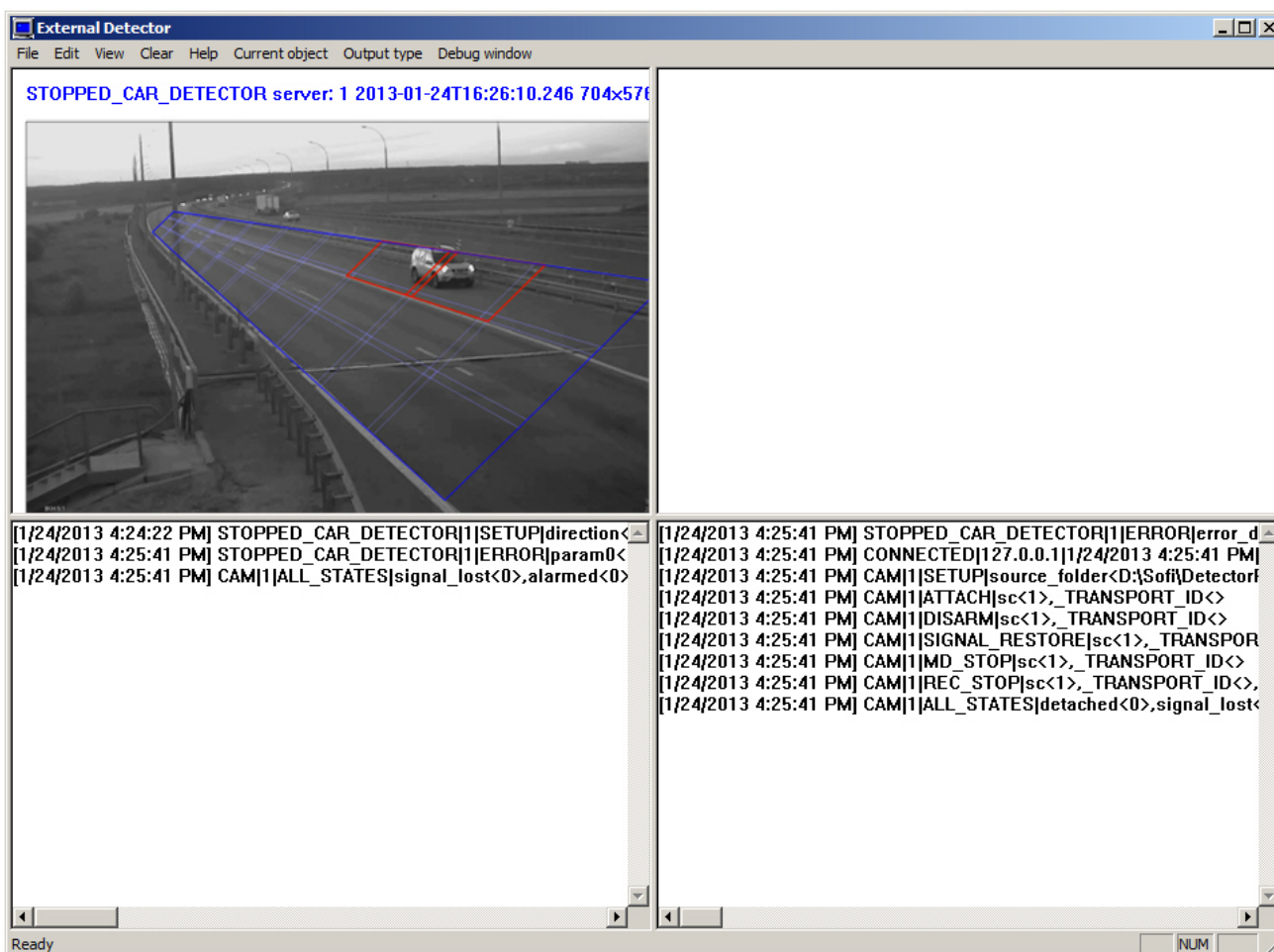
The debug window is designed to control events received from detections registered in the system. Besides, the function of displaying the detection area above the video image received from camera is available in the debug window.

### 7.2 Start the debug window

Start of the debug window is performed from the Windows task bar. Double click the left mouse button on the  icon to start the debug window.

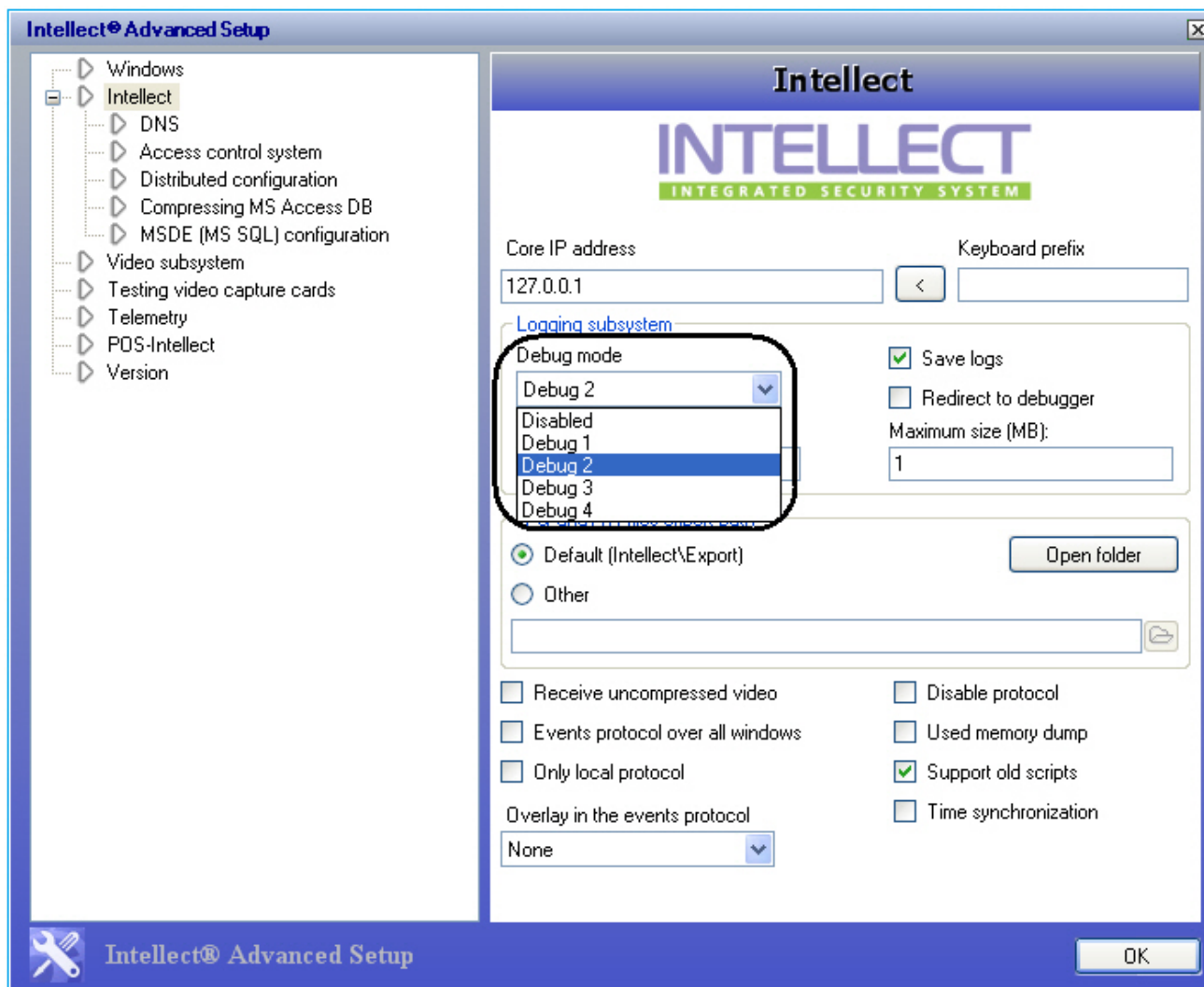


As a result the External detector window is displayed.



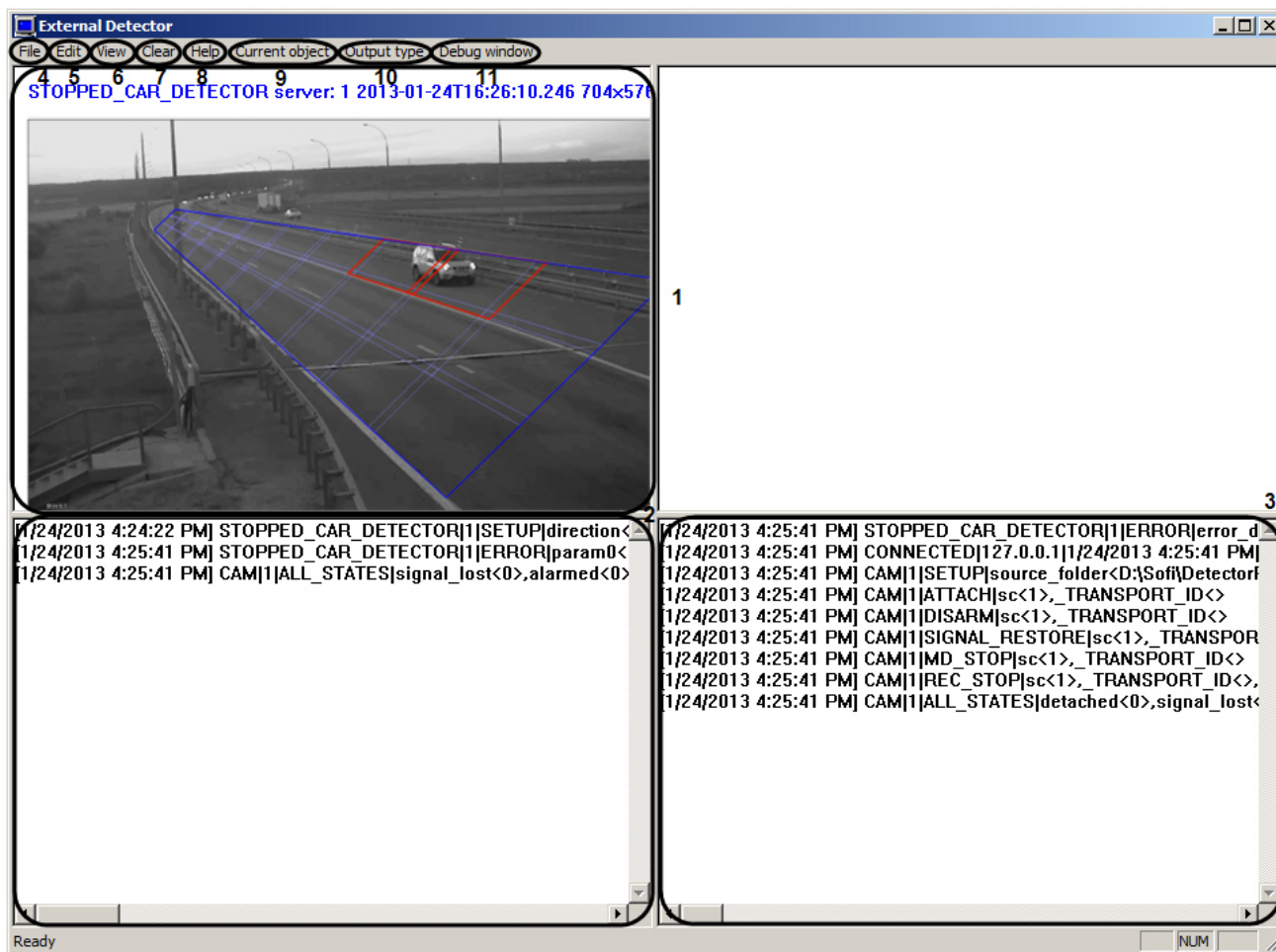
#### Attention!

Start the debug window is possible only if the **Debug** mode is enabled with the help of *Intellect Advanced Setup* utility.



### 7.3 Interface of debug window

The debug window contains the interface components described in the following table.



No	Name	Description
1	Preview area	Element is designed for displaying the detection area above the video image. Besides detection area the specific detection settings are displayed: <ul style="list-style-type: none"> <li>• the person size for queue length detection and people counter detection;</li> <li>• detection zones for stopped vehicle detection.</li> </ul>
2	Area of Events viewing from detectors	Events from detectors registered in the system are displayed in this area.
3	Area of system events viewing	All system events except events from detectors are displayed in this area.
4	<b>File</b> menu	Access to the Exit function.
5	<b>Edit</b> menu	Access to the operations with text.
6	<b>View</b> menu	Access to the function of displaying and hiding the status bar.
7	<b>Clear</b> button	Clear areas of events viewing.
8	<b>Help</b> menu	Access to the information about program.

9	<b>Current object</b> menu	Selection of a detector settings of which are displayed in the preview area. The detector selection have an impact on the camera from which the video stream is used for displaying in the preview window.
---	----------------------------	--

## 8 Appendix 2. Commands, requests and events of the Detector Pack software modules

### 8.1 Events of the software modules

All events from the Detector Pack software modules come from the **CAM** object, which corresponds to the **Camera** system object.

Events of software modules can be used in *Intellect* scripts to launch the procedures when a corresponding event occurs (see [Programming Guide](#)).

Event format:

```
CAM|"_id_"|"_command_"|"_parameters_"
```

Events description:

Event	Description	Software module used
NEW_OBJECT	The appearance of a new track (object)	Neurotracker
OBJECT_LOST	The disappearance of a track (object)	Neurotracker
CURRENT_OBJECTS	Event for the GET_STATUS command	Neurotracker

List of event parameters:

Parameters	Parameters description
cam<>	id of the software module that generates the event
fraction<>	Millisecond timestamp of the event
new_id<>	id of a new track (object)
owner<>	Server name
date<>	Date of event
guid_pk<>	id of event (randomly generated for each event)
core_global<>	When the event is generated, all CORE objects in the distributed system are notified
time<>	Time of event
lost_id<>	id of a disappeared track (object)
y.0<>...	0 - serial number of a track (object) in the event, <> - <b>y</b> coordinates
x.0<>...	0 - serial number of a track (object) in the event, <> - <b>x</b> coordinates
id.0<>...	0 - serial number of a track (object) in the event, <> - object id
id.count<>	Current number of tracks (objects)

## 8.2 Commands of the software modules

Commands of the software modules can be used in *Intellect* scripts (see [Programming Guide](#)) or sent directly to the kernel using *IIDK* (see [Intellect Integration Developer Kit \(IIDK\)](#)).

Commands description:

Command	Description	Software module used
GET_STATUS	Generates the events for each detected object. These events contain a unique object ID, relative x and y coordinates (if the upper left corner is 0.0, and the lower right corner is 100,100), date/time, etc.	Neurotracker

Example of the GET\_STATUS command:

```
DoReact (NEURO_TRACKER|1|GET_STATUS);
```