

Detectors Pack

Detector Pack. User Guide

Last update 01/04/2022

Table of contents

1	Introduction	7
1.1	General information.....	7
1.2	The purpose of the document.....	7
1.3	Purpose of the Intellect detector pack	7
2	Software and hardware requirements.....	8
2.1	General requirements for base computers and operating system	8
2.2	Hardware and software requirements for neural analytics.....	8
2.3	Platforms for the detection software modules operation	9
3	Installing the Intellect Detector pack.....	11
3.1	General description of Detector Pack distribution kit	11
3.2	Installation	11
3.3	Repair.....	14
3.4	Removal.....	15
3.5	Installing Detector Pack in silent mode	17
4	Configuring detection modules.....	19
4.1	General information on Neural analytics.....	19
4.1.1	Data collection requirements for neural network training.....	19
4.2	Selecting a camera video stream for the detection modules.....	21
4.3	Configuring the events archive length	21
4.4	Queue length detection	21
4.4.1	Functionality of the Queue length detection module.....	21
4.4.2	Camera requirements for the Queue length detection module.....	22
4.4.3	Configuring the Queue length detection module	22
4.5	People counter detection	28
4.5.1	Functionality of the People counter detection module.....	28
4.5.2	Camera requirements for the People counter detection module.....	28
4.5.3	Configuring the People counter detection module	29
4.6	Stopped vehicle detection.....	33
4.6.1	Functionality of the Stopped vehicle detection module	33
4.6.2	Camera requirements for the Stopped vehicle detection module.....	33
4.6.3	Configuring the Stopped vehicle detection module	33

4.6.3.1	Licensing the Stopped vehicle detection module.....	33
4.6.3.2	Configuring the Stopped vehicle detection module.....	34
4.7	Detection of light indication control.....	37
4.7.1	Functionality of the Detection of light indication control module.....	37
4.7.2	Camera requirements for the Detection of light indication control module.....	37
4.7.3	Configuring the Detection of light indication control module.....	38
4.8	Heat map detection.....	41
4.8.1	Functionality of the Heat map detection.....	41
4.8.2	Camera requirements for the Heat map detection module.....	41
4.8.3	Configuring the Heat map detection module.....	42
4.9	Detection of moving against crowd flow.....	46
4.9.1	Functionality of the Detection of moving against crowd flow.....	46
4.9.2	Camera requirements for the Detection of moving against crowd flow module.....	46
4.9.3	Configuring the Detection of moving against crowd flow module.....	47
4.9.3.1	Behavior of the Detection of moving against crowd flow module.....	47
4.9.3.2	Setting up the Detection of moving against crowd flow module.....	47
4.10	Sweethearting at checkout detection.....	50
4.10.1	Functionality of the Sweethearting at checkout detection module.....	50
4.10.2	Camera requirements for the Sweethearting detection module.....	51
4.10.3	Configuring the Sweethearting at checkout detection module.....	51
4.11	Barcode detection.....	55
4.11.1	Functionality of the Barcode detection.....	55
4.11.2	Camera requirements for the Barcode detection module.....	55
4.11.3	Configuring the Barcode detection module.....	55
4.12	Train detection.....	58
4.12.1	Functionality of the Train detection module.....	58
4.12.2	Camera requirements for the Train detection module.....	58
4.12.3	Configuring the Train detection module.....	58
4.13	Fire detection and Smoke detection.....	62
4.13.1	Functionality of the Fire detection and Smoke detection modules.....	62
4.13.2	Camera requirements for the Fire detection and Smoke detection modules.....	62
4.13.3	Configuring the Fire detection and Smoke detection modules.....	62
4.14	Fluid level detection.....	66
4.14.1	Functionality of the Fluid level detection module.....	66
4.14.2	Camera requirements for the Fluid level detection module.....	66

4.14.3	Configuring the Fluid level detection module	66
4.15	Neurotracker	68
4.15.1	Functionality of the Neurotracker module	68
4.15.2	Video stream and scene requirements for the Neurotracker module	69
4.15.3	Objects image requirements for neural tracker	69
4.15.4	Configuring the Neurotracker module	71
4.15.4.1	Configuring the neurotracker track counter	75
4.15.4.2	Examples of configuring neural tracker for solving typical tasks	77
4.16	Person location tracker	78
4.16.1	Functionality of the Person location tracker module	78
4.16.2	Video stream and scene requirements for Person location tracker	78
4.16.3	Objects image requirements for Person location tracker	78
4.16.4	Configuring the Person location tracker module	79
4.16.4.1	Configuring the pose detection	81
4.17	VideoIntellect embedded detector	84
4.17.1	Functionality of the VideoIntellect embedded detector module	84
4.17.2	Licensing of the VideoIntellect embedded detector module	84
4.17.3	Camera requirements for the VideoIntellect embedded detector module	85
4.17.4	Configuring the VideoIntellect embedded detector module	85
4.17.4.1	Configuring VideoIntellect detection tools	86
4.17.4.1.1	VideoIntellect abandoned objects detection	86
4.17.4.1.2	VideoIntellect prohibited zone motion detection	89
4.17.4.1.3	VideoIntellect camera state detection	92
4.17.4.1.4	VideoIntellect prohibited direction motion detection	94
4.17.4.1.5	VideoIntellect other detection	97
4.18	Neurocounter	99
4.18.1	Functionality of the Neurocounter module	99
4.18.2	Video stream and scene requirements for Neurocounter module	99
4.18.3	Objects image requirements for Neurocounter	100
4.18.4	Configuring the Neurocounter module	102
4.18.5	Example of configuring Neurocounter for solving typical task	105
4.19	Equipment detection (PPE)	106
4.19.1	Functionality of the Equipment detection (PPE) module	106
4.19.2	Video stream and scene requirements for Equipment detection tool (PPE) operation	106
4.19.3	Objects image requirements for Equipment detection tool (PPE)	107

4.19.4	Example of providing a list of valid equipment at the facility	107
4.19.5	Configuring the Equipment detection (PPE) module.....	110
4.19.6	Examples of configuring Equipment detection tool (PPE) for solving typical tasks	113
4.20	Traffic light detection	114
4.20.1	Functionality of the Traffic light detection module	114
4.20.2	Configuring the Traffic light detection module	114
5	Operating with detection modules	118
5.1	Operating the «Queue length detection» module.....	118
5.1.1	Obtaining traffic information in the area of interest.....	118
5.1.2	Generating a report on the traffic in the area of interest	118
5.1.3	Visualization of operating the Queue length detection	118
5.2	Operating the «People counter detection» module.....	119
5.2.1	Obtaining information on number of visitors.....	119
5.2.2	Generating a visitor report	119
5.2.3	Visualization of operating the People counter detection	120
5.3	Operating the «Stopped vehicle detection» module	120
5.4	Operating the Detection of light indication control module	121
5.5	Operating the «Heat map detection».....	121
5.5.1	Generating a report on the basis of Heat map detection.....	121
5.6	Operating the «Detection of moving against crowd flow».....	121
5.7	Operating the «Sweethearting detection» module.....	121
5.7.1	Generating sweethearting reports.....	122
5.8	Operating the «Barcode detection» module	122
5.9	Operating the «Train detection» module.....	122
5.10	Operating the "Fire detection" and "Smoke detection" modules	123
5.11	Operating the «Fluid level detection» module	123
5.12	Operating the Neurotracker module	125
5.13	Operating the Person location tracker module.....	125
5.14	Operating the VideoIntellect embedded detector	125
5.15	Operating the Neurocounter module	125
5.16	Operating the Equipment detection (PPE)	126
6	Appendix 1. Debug window	127
6.1	General information	127
6.2	Start the debug window	127

6.3	Interface of debug window	128
7	Appendix 2. Commands, requests and events of the Detector Pack software modules.....	131
7.1	Events of the software modules.....	131
7.2	Commands of the software modules.....	132

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* 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. **Detection of light indication control.**
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.**
15. **VideoIntellect embedded detector.**
16. **Neurocounter.**
17. **Equipment detection (PPE).**
18. **Traffic light detection.**

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

2 Software and hardware requirements

On this page:

- [General requirements for base computers and operating system](#)
- [Hardware and software requirements for neural analytics](#)

2.1 General requirements for base computers and operating system

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 neural analysis (see [General information on Neural analytics](#)), the operation device can be set manually.

2.2 Hardware and software requirements for neural analytics

The following requirements are imposed on the software and hardware platform when the neural network analytics is used:

1. Due to the NVIDIA SDK specific features, the neural analytics can operate only on Windows Server 2019 OS and Windows 10 OS.
2. The neural network analytics supports the following devices: CPU, GPU NVIDIA, 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. Several Intel HDDL devices can be used on one Server if their revisions match. For Intel HDDL to work on a computer with AMD CPU, it is required that the OpenVino toolkit be installed according to the instructions on the [site](#).

3. If CPU or Intel GPU is used for the neural network analytics operation, then the following requirements should be taken into account:
 - a. support for the following CPUs:
 - i. 6th-10th Generation Intel® Core™ processors;
 - ii. Intel® Xeon® v5 family;
 - iii. Intel® Xeon® v6 family;
 - iv. Intel® Movidius™ Neural Compute Stick;
 - v. Intel® Neural Compute Stick 2;

- vi. Intel® Vision Accelerator Design with Intel® Movidius™ VPUs;
 - b. the OpenVino toolkit should support the Intel CPU being used (for more information see <https://software.intel.com/content/www/us/en/develop/tools/openvino-toolkit/system-requirements.html>).
4. NVIDIA GeForce 1050 Ti GPU or newer. GPU requirements:
- a. at least 2 GB of memory;
 - b. Compute Capability 3.5 - 8.6.

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](#)).

2.3 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	✓	✓
Detection of light indication control	✓	✓
Sweethearting detection	✗	✓
Barcode detection	✓	✗

Fluid level detection		
Neurotracker		
Person location tracker		
Videointellect embedded detector		
Neurocounter		
Equipment detection (PPE)		
Traffic light detection		

 **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](#).

3 Installing the Intellect Detector pack

3.1 General description of Detector Pack distribution kit

Detector Pack is supplied as a software installation package (distribution kit). The current version of the distribution kit can be downloaded from the official [AxxonSoft](#) website.

The distribution kit contains all the necessary software components for installing the *Detector Pack* software package on a base computer.

The distribution kit allows you to install, restore and remove the *Detector Pack* software package.

⚠ Attention!

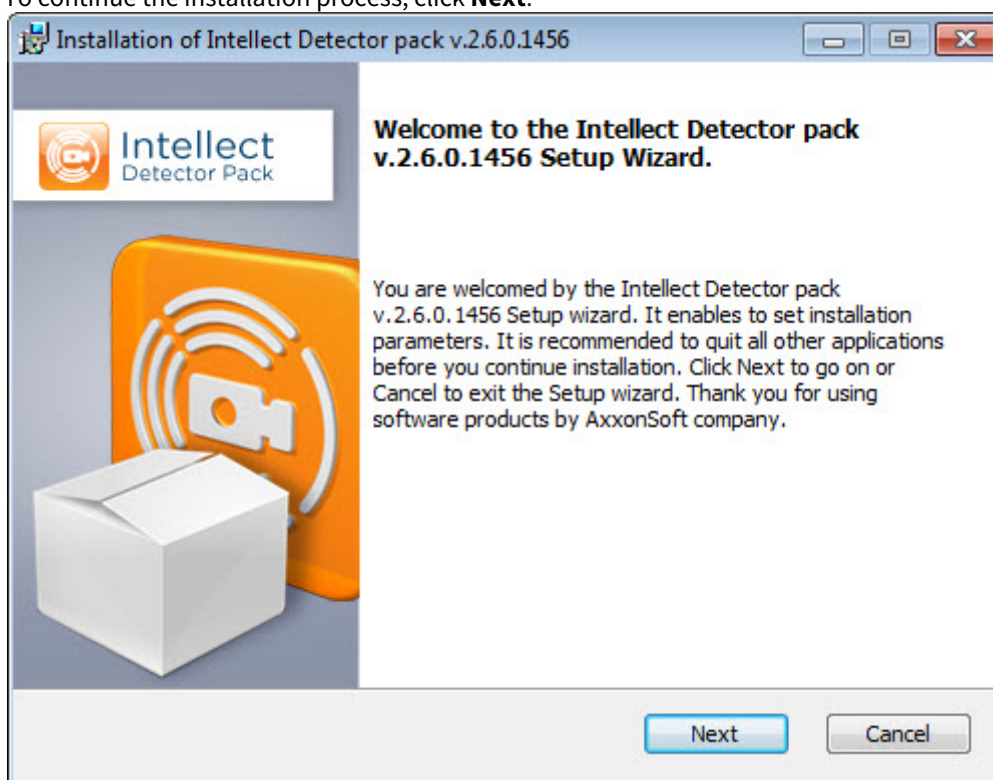
- Prior to installing, restoring or removing the *Detector Pack* software package, the *Intellect* operation should be shut down.
- Administrator rights are required for installing, restoring or removing *Detector Pack*.

3.2 Installation

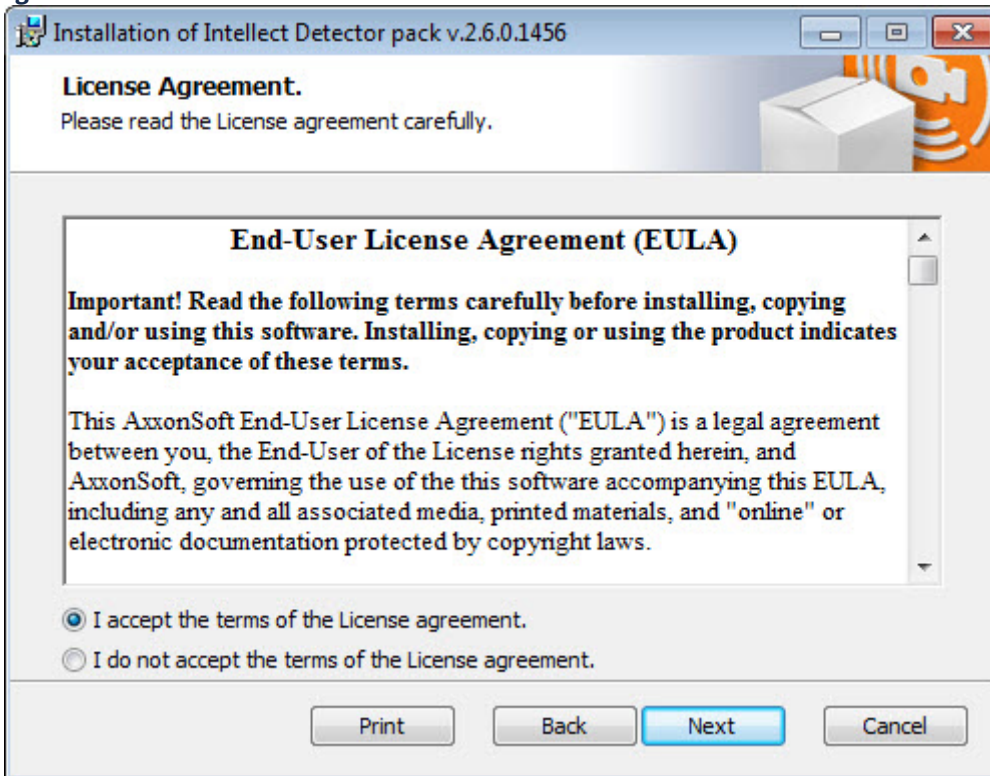
The *Intellect Detector Pack* software is installed as a part of the *Intellect* software. Information about compatibility of the *Intellect* software versions and *Intellect Detector Pack* is available at [General information about product releases and versions compatibility](#).

To install *Detector Pack*, do the following:

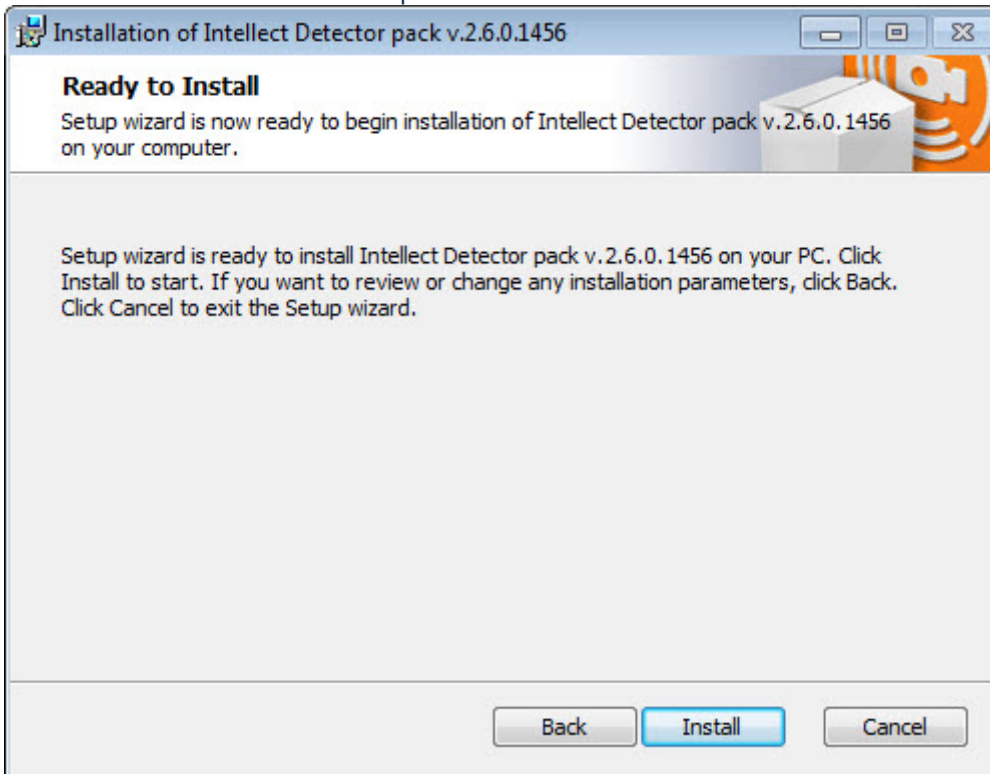
1. In the root directory of the distribution package, run the setup.exe executable file
2. To continue the installation process, click **Next**.



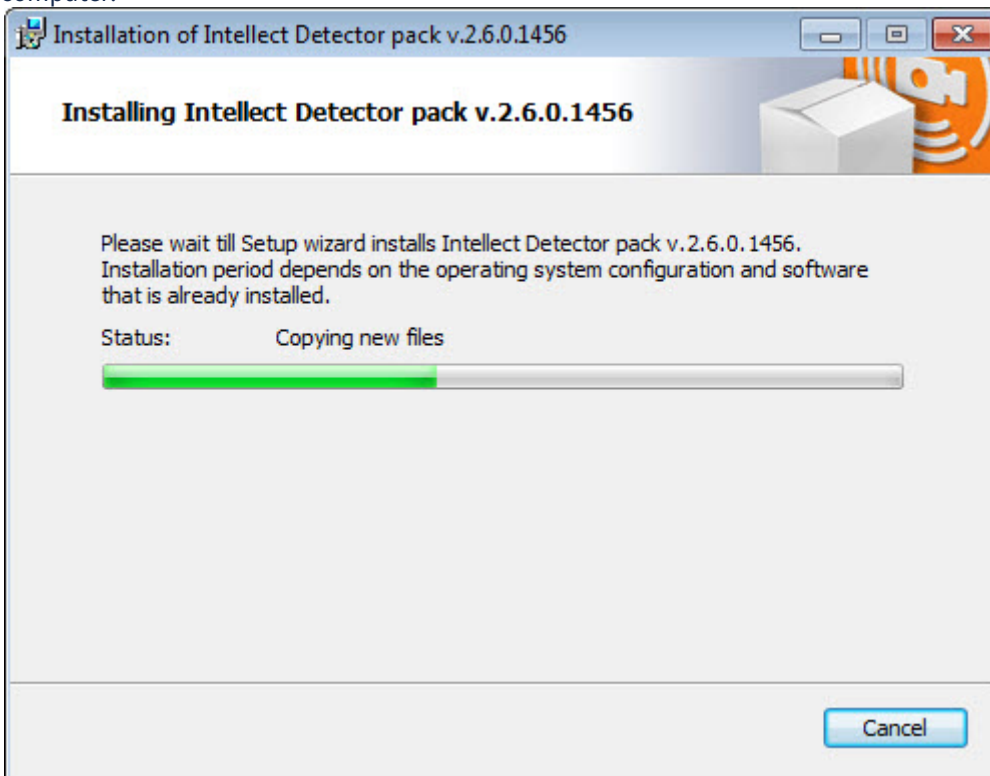
3. Read the terms of the license agreement carefully. Then set the radio button to **I accept the terms of the License agreement** and click **Next**.



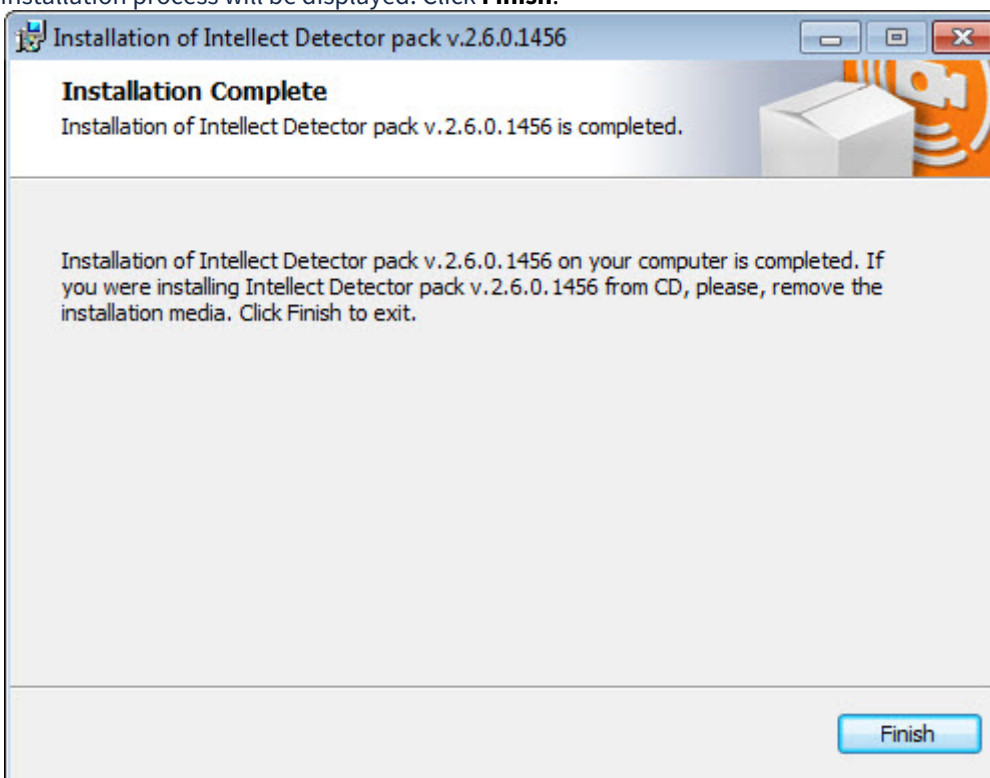
4. Click **Install** to start the installation process.



5. As a result, the necessary components of the *Detector Pack* software package will be copied to hard drive of your computer.



6. After all software components are successfully copied on your hard drive, the message about the completion of the installation process will be displayed. Click **Finish**.



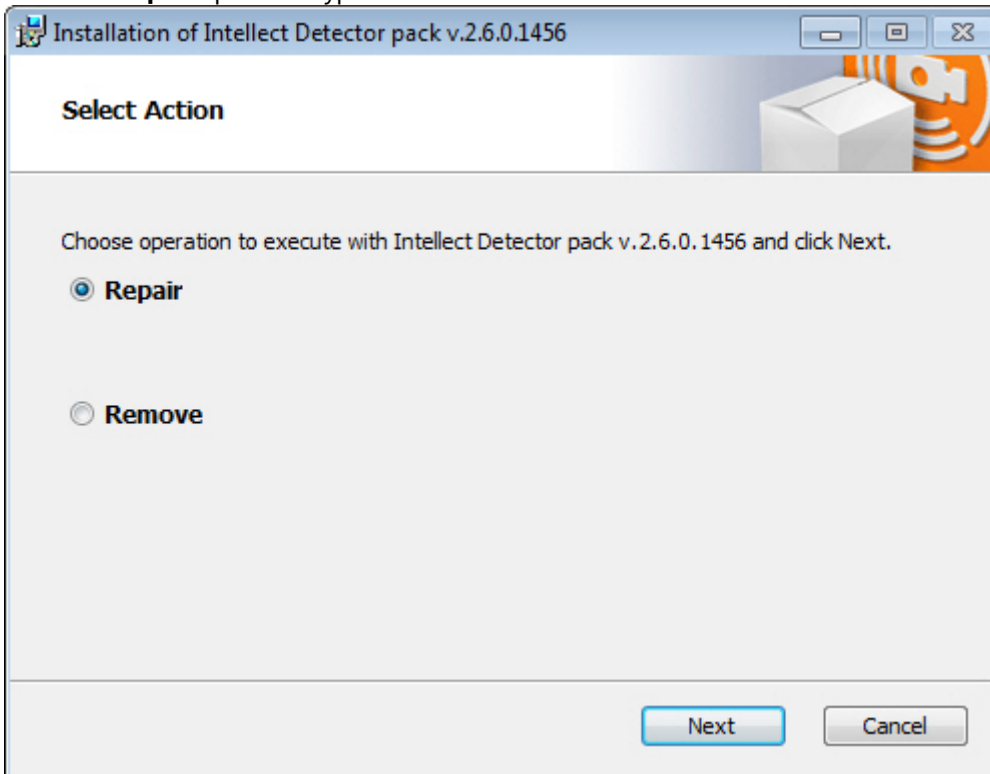
The *Intellect Detector pack* installation in complete.

3.3 Repair

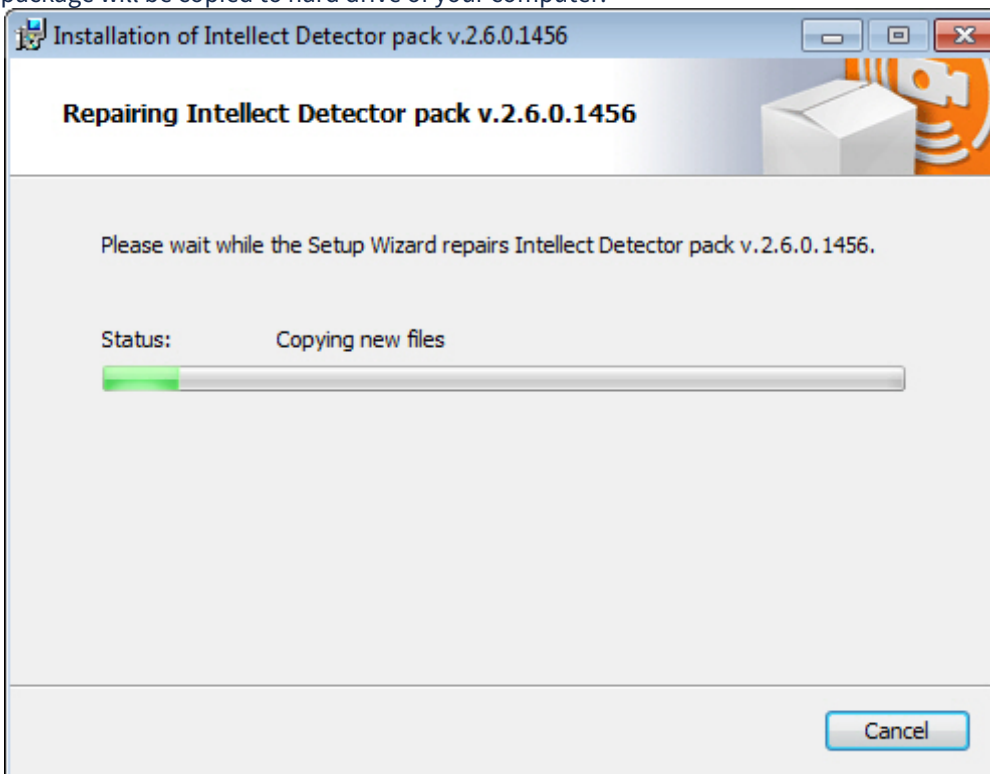
The repair mode is used if *Detector Pack* software components need to be re-installed.

To repair the *Detector Pack* software, do the following:

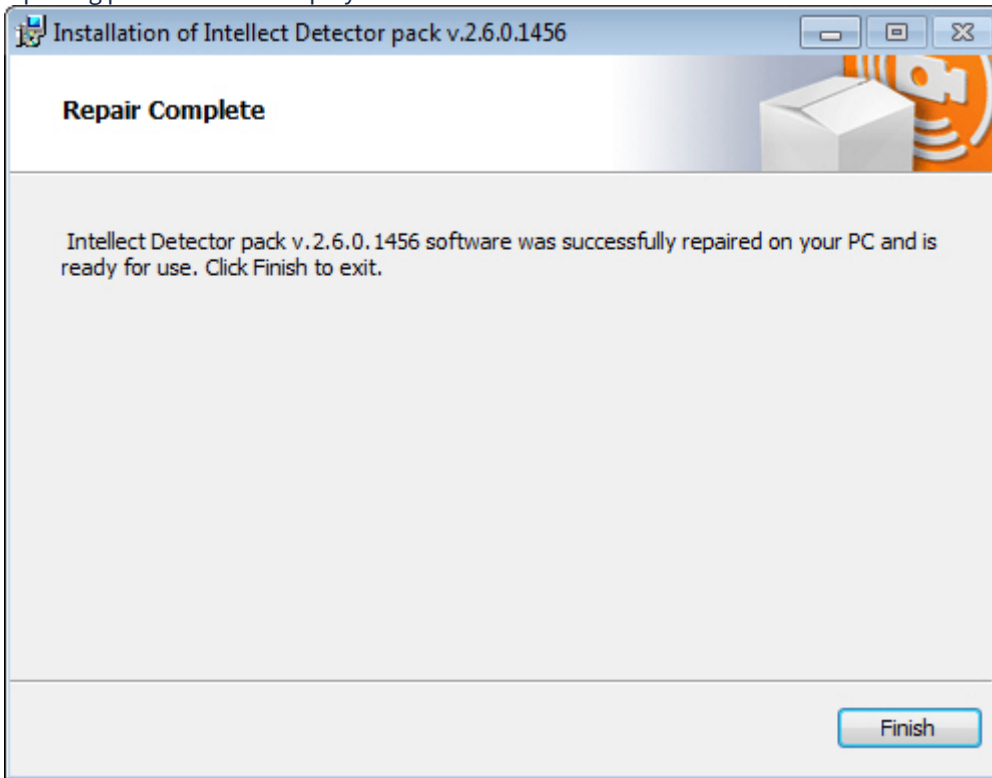
1. In the root directory of the distribution kit, run the setup.exe file.
2. Select the **Repair** operation type and click **Next**.



3. As a result, the installed components will be checked and the necessary components of the *Detector Pack* software package will be copied to hard drive of your computer.



4. After all software components are successfully copied on your hard drive, the message about the completion of the repairing process will be displayed. Click **Finish**.



The *Intellect Detector Pack* repair is complete.

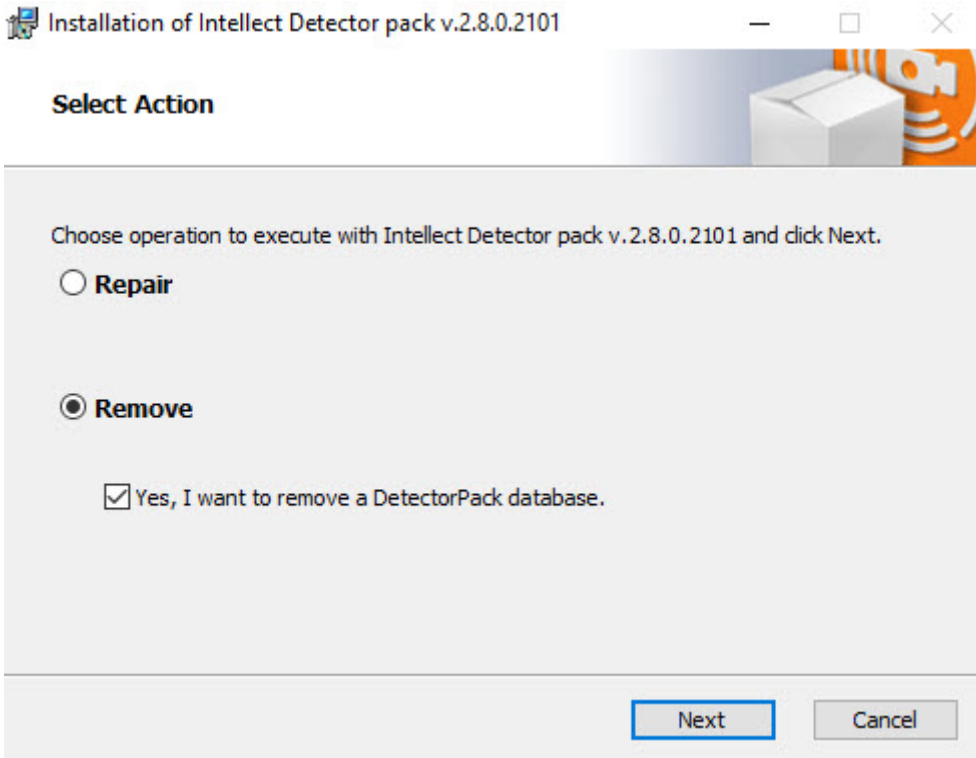
3.4 Removal

To remove the *Detector Pack* software, do the following:

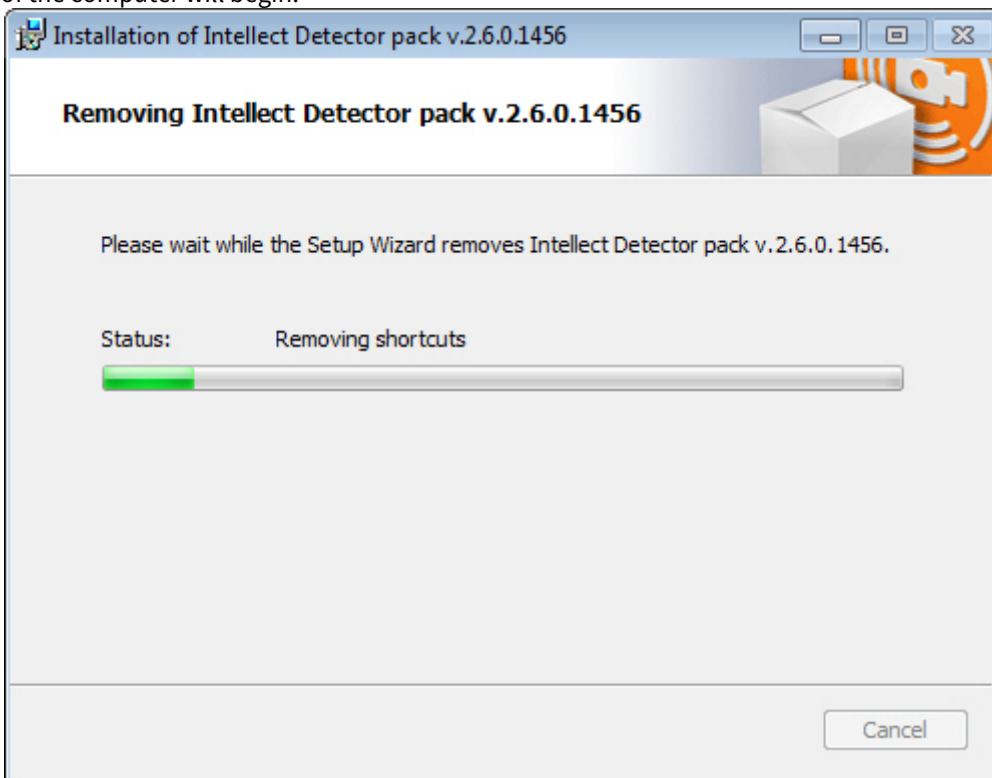
1. In the root directory of the distribution kit, run the setup.exe file.
2. Select the **Remove** operation type and click **Next**.

Note

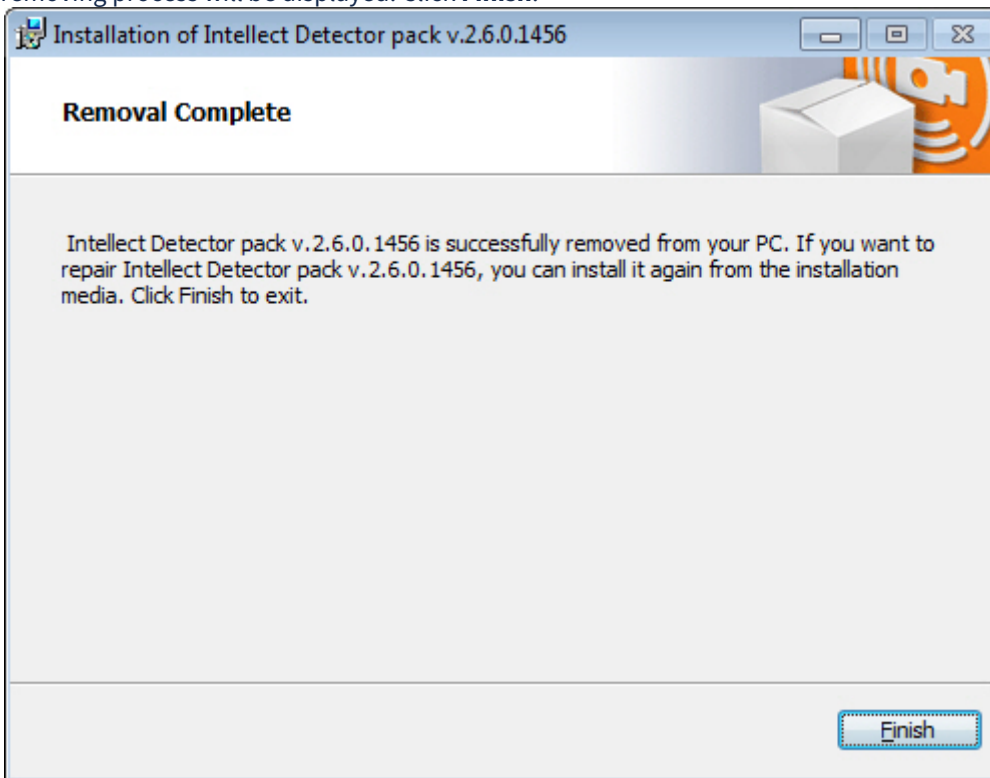
If it is also necessary to delete the *Detector Pack* database, set the **Yes, I want to remove a DetectorPack database** check box.



3. As a result, the process of removing the installed components of the *Detector Pack* software package from the hard drive of the computer will begin.



- After all software components are successfully removed from hard drive, the message about the completion of the removing process will be displayed. Click **Finish**.



The *Intellect Detector Pack* removal is complete.

3.5 Installing Detector Pack in silent mode

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

- 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\vc_redist_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.

- 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.

4 Configuring detection modules

4.1 General information on Neural analytics

✓ Software and hardware requirements

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.
2. **AI Smoke and Fire detection** (see [Fire detection and Smoke detection](#)).
Neural network detects fire and smoke in FoV.
3. **AI Person location tracker** (see [Person location tracker](#)).
AI-powered Posture Detection captures specific human poses that may represent a security threat.
4. **Neurocounter** (see [Neurocounter](#)).
Neural network counts the number of objects in a given area.
5. **Sweethearting detection** (see [Sweethearting at checkout detection](#)).
Neural network detects the theft of goods by cashiers who do not scan some items barcodes at the checkout.
6. **Equipment detection** (see [Equipment detection \(PPE\)](#)).
Neural network divides the human body into segments, detects the equipment (PPE) on a given body segment, and checks its condition.

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

ⓘ Notes

The file names for each neural network model depend on the device on which the neural network will operate:

- If you intend to use a CPU or Intel GPU (integrated video core), 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 a GPU (NVIDIA graphics processor), 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 NCS or Intel HDDL, then the name of the neural network file should contain the **_movidius** substring at the end. For example, **test1_movidius.ann**.
- 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.

Before you start setting up the AI-based detection tools, you should contact the [AxxonSoft](#) technical support and request the model files of the trained neural networks. Technical support specialists will request the required data (see [Data collection requirements for neural network training](#)) and then provide the files for each neural network model. These files should be distributed to all Servers where it is planned to use the detection tools.

⚠ Attention!

The startup (initialization) time of each neural network on NVIDIA GPU can take 2-3 minutes, depending on the used neural network model. Until initialization is complete, no events will be received from detection tools.

4.1.1 Data collection requirements for neural network training

To train neural networks, it is necessary to collect and submit to AxxonSoft video recordings and images from your actual cameras taken in the same resolution and under the same conditions as in your future application.

For example, if your neural network is intended to analyze outdoor video feeds, your footage must contain all range of weather conditions (sun, rain, snow, fog, etc.) in different times of day (daytime, twilight, night).

General requirements for collected data:

- When collecting video recordings and images, specific requirements for object images, scene, angle, illumination and video stream are met for those detection tools that you plan to use (see [Configuring detection modules](#)).
- If it is required to train the neural network in different conditions of time of day, lighting, angle, object types or weather, then the video material should be collected in equal shares for each condition, that is, it should be balanced.

Note

Example. It is necessary to detect a person in the surveillance area at night and during the day.

Data collected correctly:

- 4 video recordings of the surveillance area, each 5 minutes long;
- the object of interest appears in the frame in each video fragment;
- 2 fragments were recorded in night conditions, 2 — in daytime conditions.

Data collected incorrectly:

- 3 video recordings of the surveillance area, each 5 minutes long;
- the object of interest appears in the frame in each video fragment;
- 2 fragments were recorded in night conditions, 1 — in daytime conditions.

Extra requirements for video footage for each neural analytics tool are listed in the following table:

Tool	Requirements
Neural Filter	No less than 1000 frames containing objects of interest in given scene conditions, and the same amount of footage containing no objects (background footage).
Neural Tracker	3 to 5 minutes of video containing objects of interest in given scene conditions. The more the number and variability of the situations in the scene, the better.
Posture detection tools	10 seconds of video of a scene with no persons. No less than 100 different persons in given scene conditions. Attention! Different conditions mean, among others, different postures of an individual in scene (tilting, different limbs patterns, etc.).
Equipment detection tool (PPE)	A list of all reference equipment with examples should be collected from the object and agreed with the analytics manufacturer (see Example of providing a list of valid equipment at the facility). Several video recordings 3-5 minutes each with personnel in the given scene conditions. Personnel should move and change posture in the collected video recordings, as well as remove and put on equipment at intervals of 30 seconds. Since the Equipment detection tool (PPE) is designed for artificial constant lighting, video recordings in other lighting conditions are not required.
Food recognition*	Images of at least 80% of the actual menu items should be provided. Each menu item requires 20 to 40 images shot in different conditions.

If the above requirements for the collection of data transmitted for training the neural network model are met, and if the neural network is operated in conditions that are as similar as possible to the conditions in which the material for its training was collected, then the overall accuracy** of neural network analytics is guaranteed from 90% to 97% and the percentage of false positives is 5-7%. For general networks***, an overall accuracy of 80-95% and a false positive rate of 5-20% are guaranteed.

Note

- * Will be available in future versions of *Intellect* software.
- ** Accuracy is indicated for a neural network model, which was trained under operating conditions.
- *** A general network is a network that was not trained under operating conditions.

The requirements may be changed or added to at any time.

4.2 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](#)).

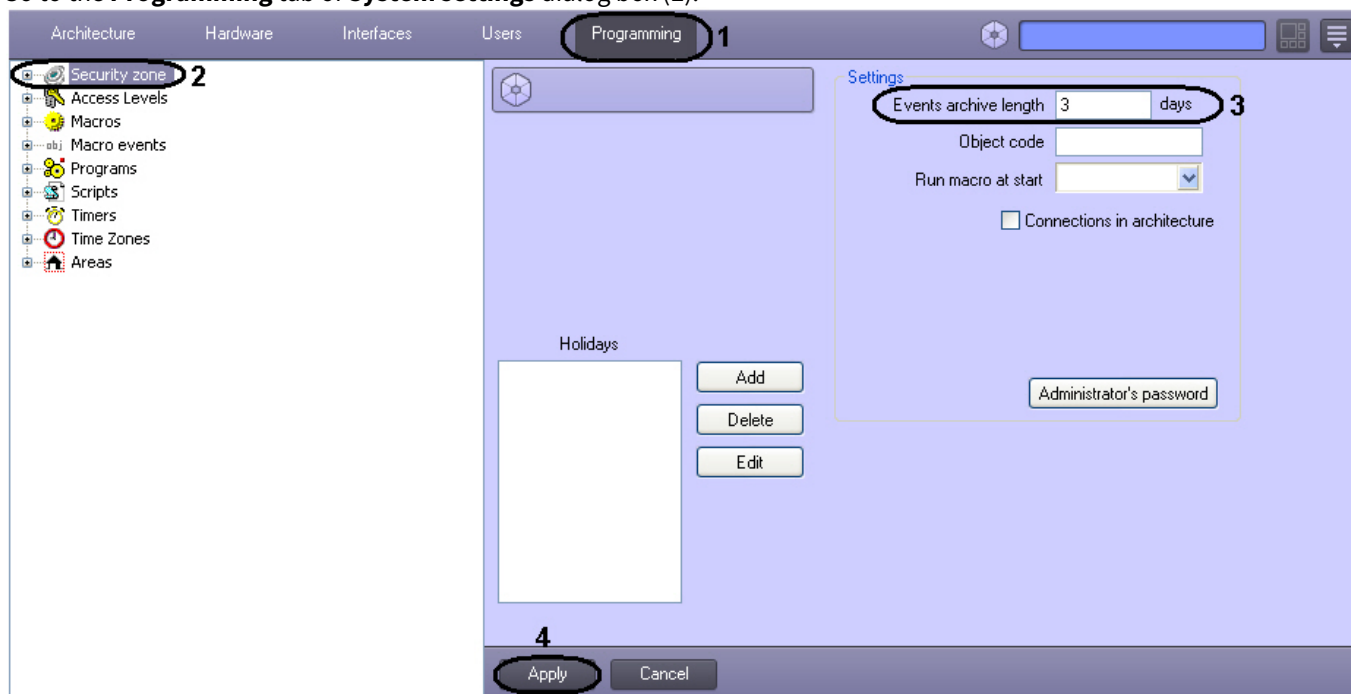
Note

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.

4.3 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.

4.4 Queue length detection

4.4.1 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.

4.4.2 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"> Resolution: 720x576 (CIF4), using of 360x288 (CIF1) is acceptable; oversize images are reduces until CIF4. Fps: not less than 6 Color: analytics works with grey and color images. Camera must be rigidly fixed.
Lighting	<ul style="list-style-type: none"> 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. Sharp changes of lighting can lead to short-time invalid analytics working.
Scene and camera angle	<ul style="list-style-type: none"> The best position – camera "looks" to the scene vertically down. The better this requirement, the carefully the received estimation. Sizes of camera field of view: 3x3m is minimal (6x6 people), 4x4m is optimal (8x8 people), 8x8m is maximal (16x16 people). Background is static and is not changed sharply. Analytics can work inappropriately on specular surfaces and in case of sharp shadows from moved objects. Analytics can work inappropriately in case of in the camera field of view there are periodic movements of background objects (trees, working TV, etc.)
Objects image	<ul style="list-style-type: none"> Image quality: the image is to be clear, without visible defects from reducing procedure. 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%

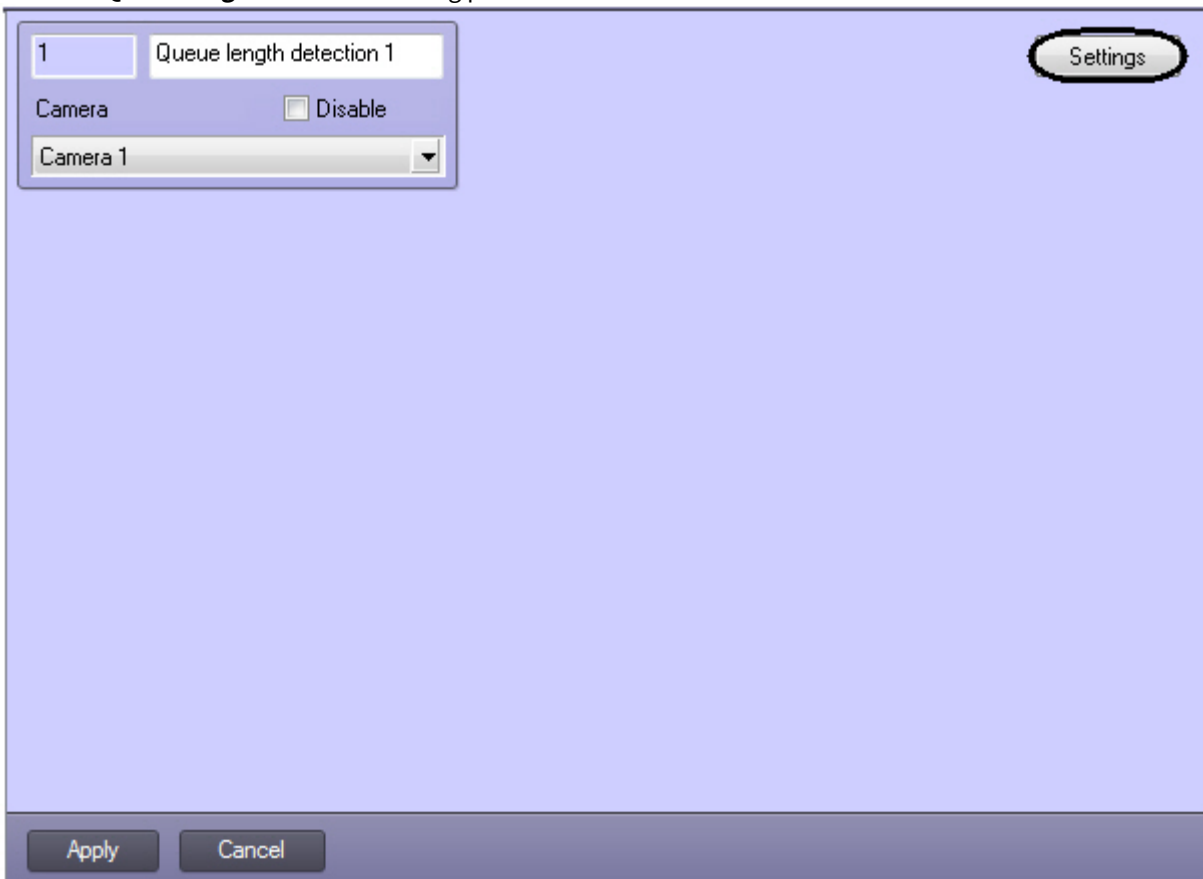
4.4.3 Configuring the Queue length detection module

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.

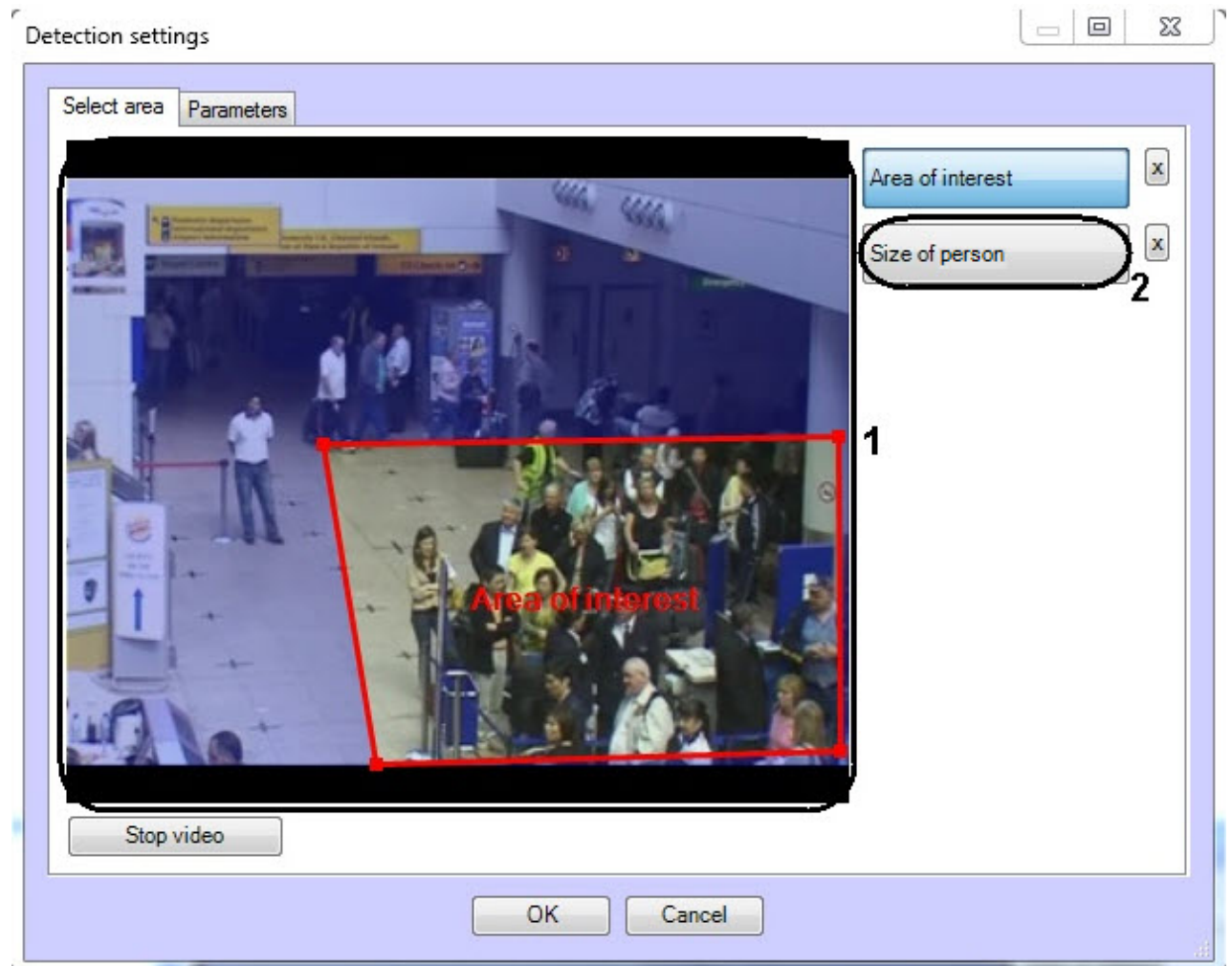


3. Specify the area of interest and the approximate size of a person in the video image:

- 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.

Note

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



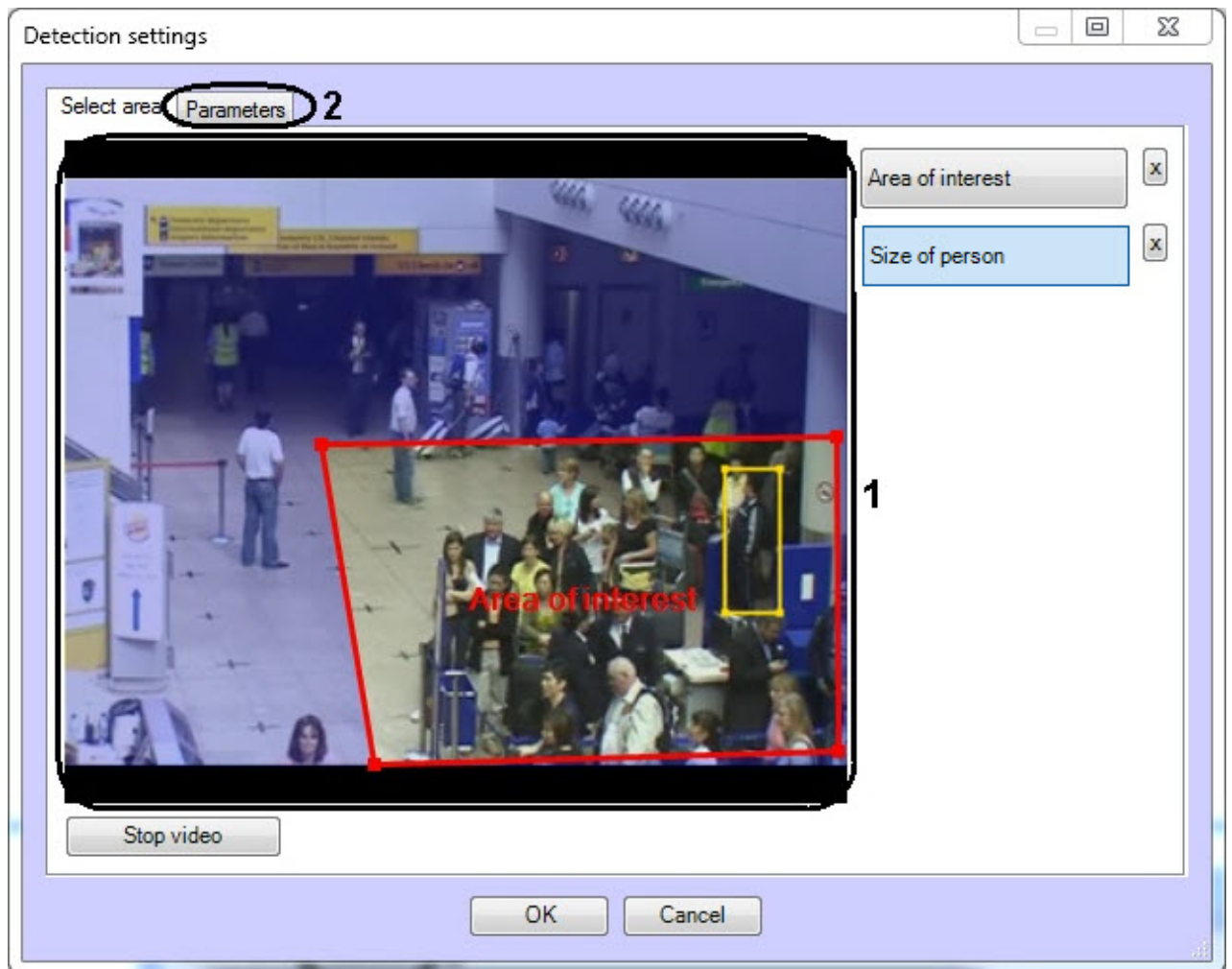
⚠ Attention!

Setting the **Area of interest** is a mandatory requirement for the detection module operation.

- d. Click **Size of person (2)**.
- e. In the captured video image **(1)** specify the approximate size of a person. To do this use the left mouse button to specify a rectangular area **(1)**.

📘 Note

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



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 occupancy measurement, 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 often the occupancy is measured, the greater the load on the system.

- 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 **Queue size, person** field, enter the minimum number of people in a queue required for triggering the detection tool (4).
- f. In the **Frames processed per second (0.016-100)** field, enter the number of frames per second that the detection tool will process (5). The higher the value, the more accurate the detection tool's operation, but the higher the load on the processor.

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 (6).

Note

To return to the settings panel of the **Queue length detection** object without saving changes, click **Cancel (7)**.

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

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

4.5 People counter detection

4.5.1 Functionality of the People counter detection module

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

- Count visitors in an observed area.
- Record events about visitor entries into the observed area to a database.
- Record events about visitor exits from the observed area to a database.
- 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.

4.5.2 Camera requirements for the People counter detection module

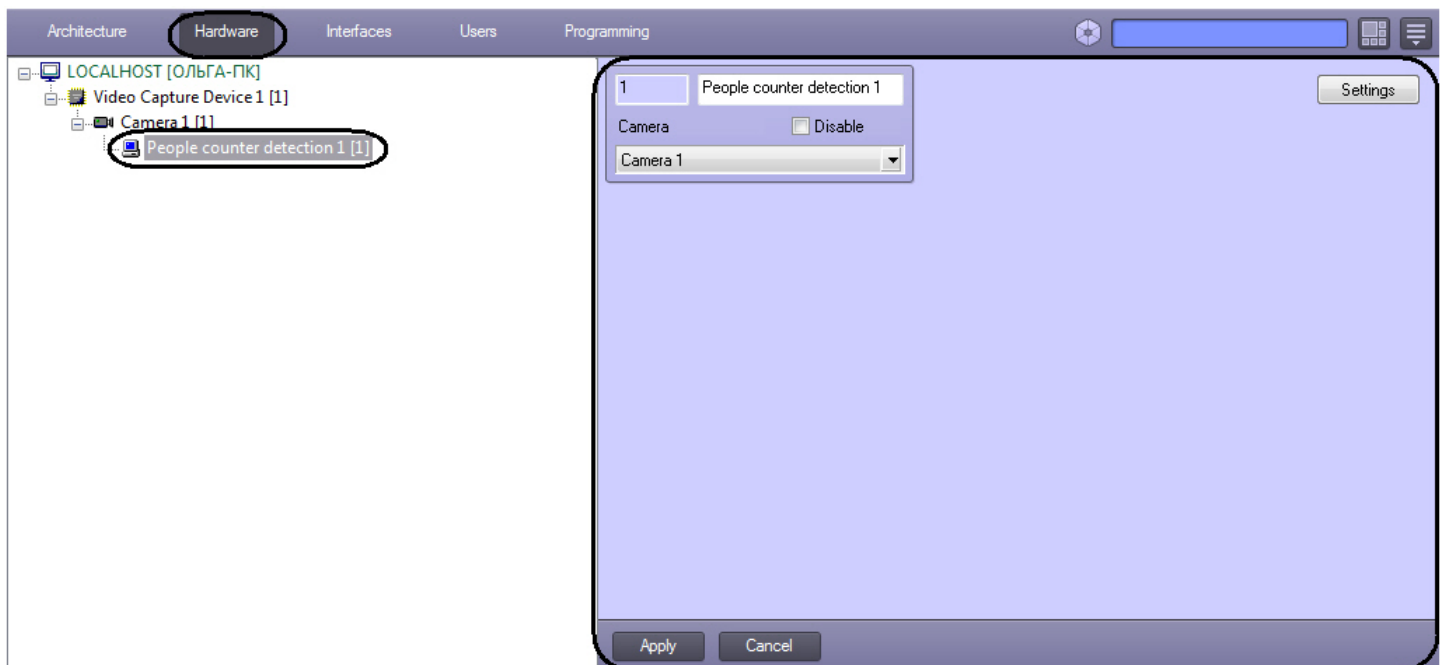
The requirements for the cameras that will work with the **People counter detection** module are listed in the following table.

Camera	<ul style="list-style-type: none"> Resolution: 720x576 (CIF4), using of 360x288 (CIF1) is acceptable. Resolution zoom-in over CIF4 is not improve the quality of recognizing procedure. Fps: 25. Color: only color camera can be in use. Camera must be rigidly fixed.
Lighting	<ul style="list-style-type: none"> 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. Sharp changes of lighting can lead to short-time invalid analytics working.
Scene and camera angle	<ul style="list-style-type: none"> The best position – camera "looks" to the scene vertically down. The better this requirement, the carefully the received estimation. Sizes of camera field of view: 2x2m is minimal, 4x4m is optimal (8x8 people). Background is static and is not changed sharply. In the recognized are there no moving objects except of people. Analytics can work inappropriately on specular surfaces and in case of sharp shadows from moved objects. Analytics can work inappropriately in case of in the camera field of view there are periodic movements of background objects (trees, working TV, etc.)

	<ul style="list-style-type: none"> • People occulting by static objects is to be minimal (by columns, trees etc.).
Objects image	<ul style="list-style-type: none"> • Image quality: the image is to be clear, without visible defects from reducing procedure. • 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%.
Other:	<ul style="list-style-type: none"> • People shouldn't move by continuous flow, by groups by several people are counted properly.

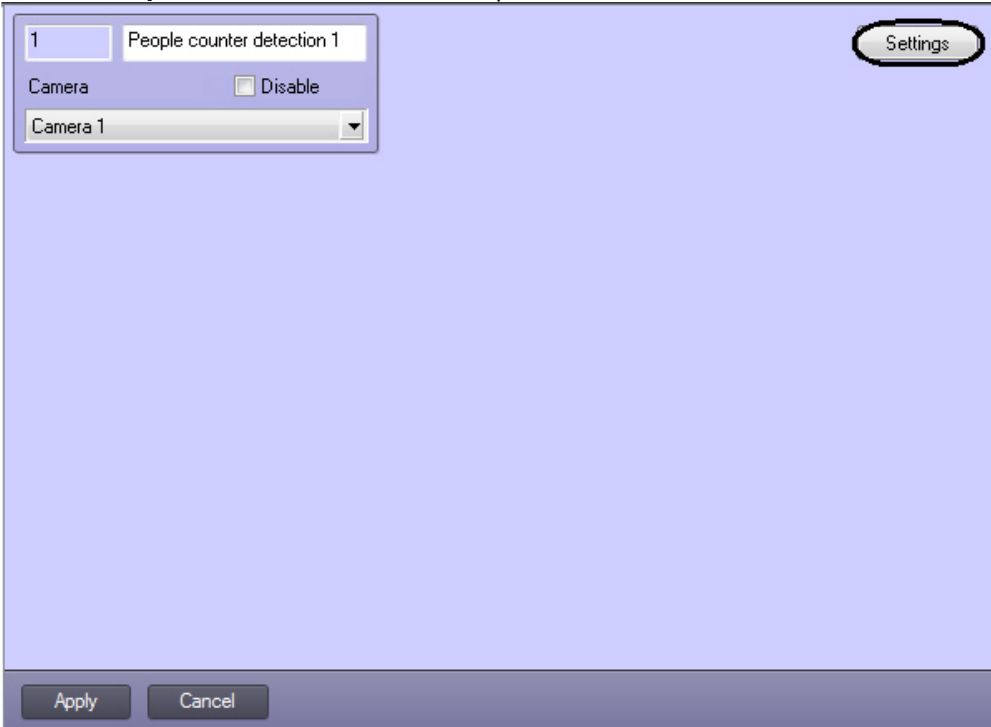
4.5.3 Configuring the People counter detection module

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.

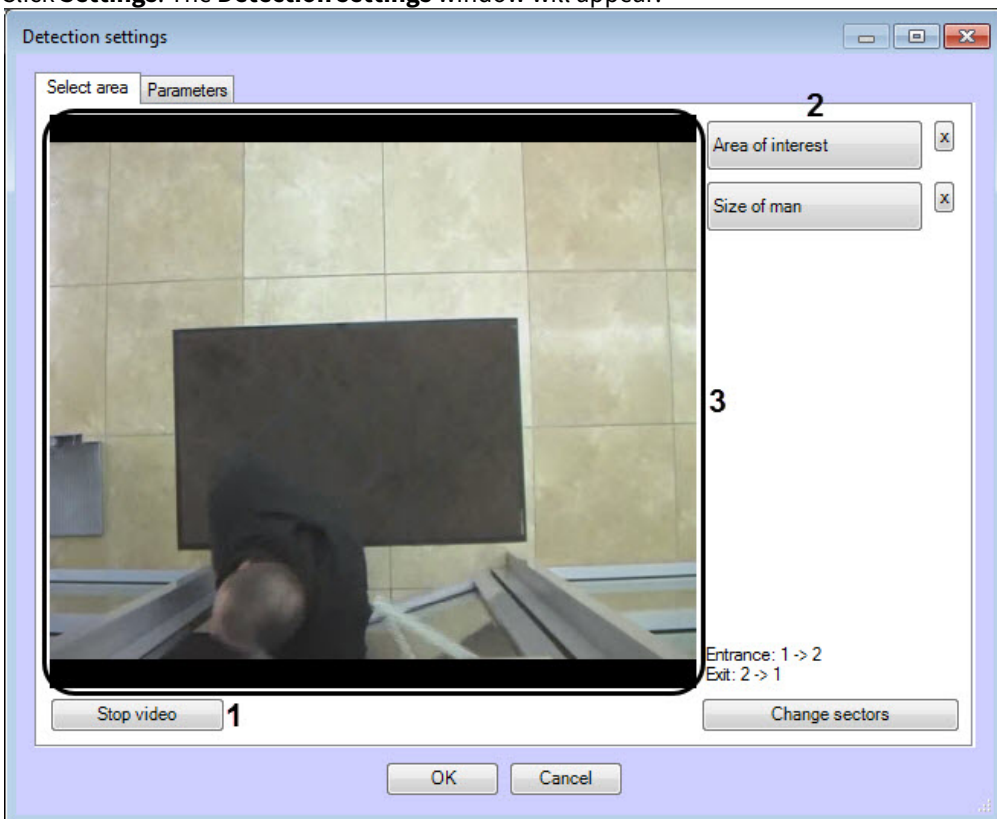


The **People counter detection** module is set up as follows:

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




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



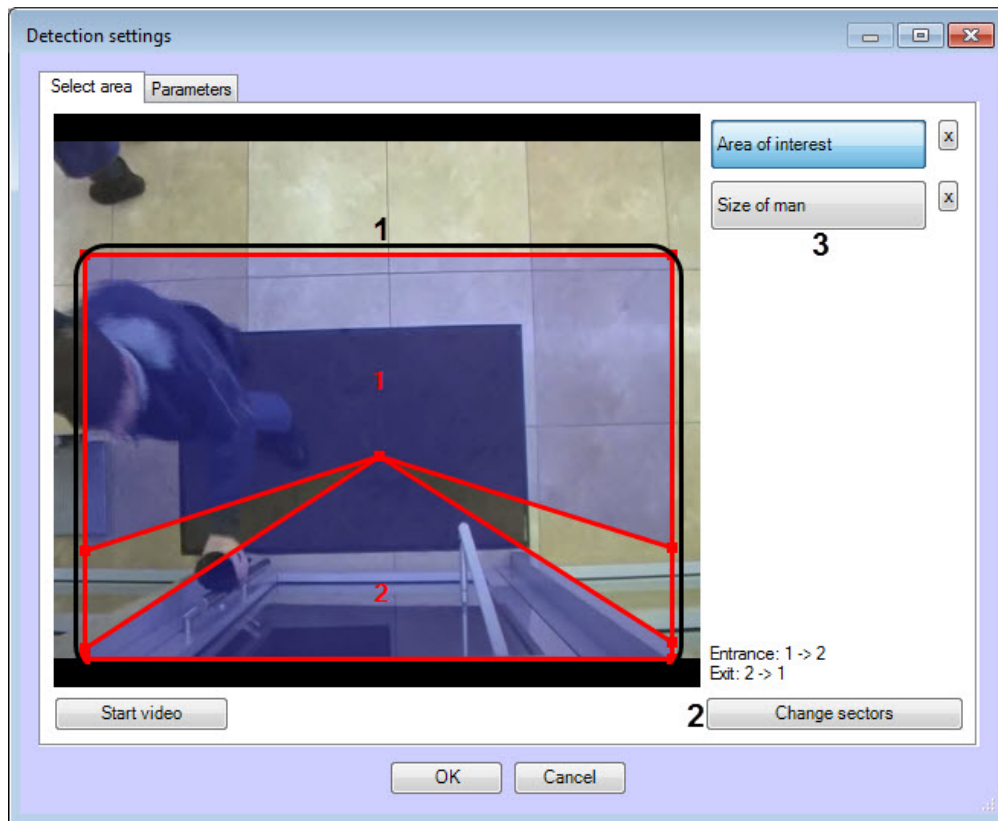
3. Specify the area of interest and the approximate size of people in the video image:
 - a. Click **Stop video** to capture the video image (1).
 - b. Click **Area of interest** (2).
 - c. Using the left mouse button select the four corners of the area on the captured video image (3) to be analyzed. 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.

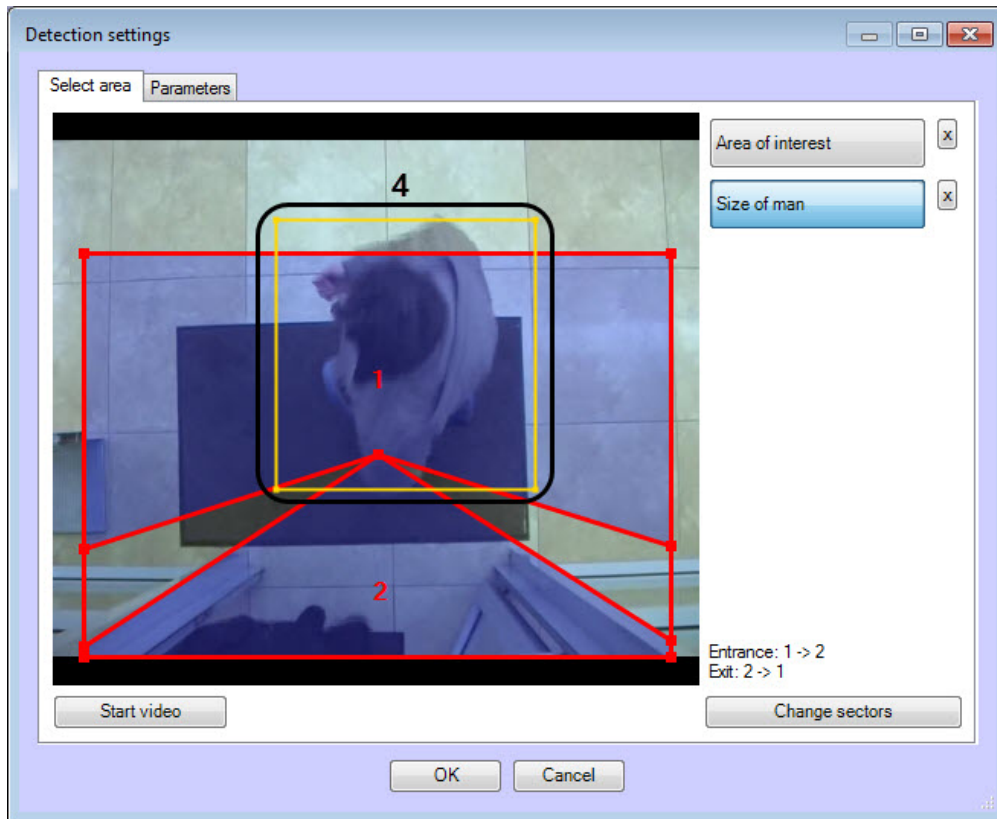
The area of interest is divided into two sections - 1 and 2. If an object moves from sector 1 to sector 2, it is logged as the entry of a visitor; if the visitor moves from sector 2 to sector 1, it is logged as an exit.



- 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).

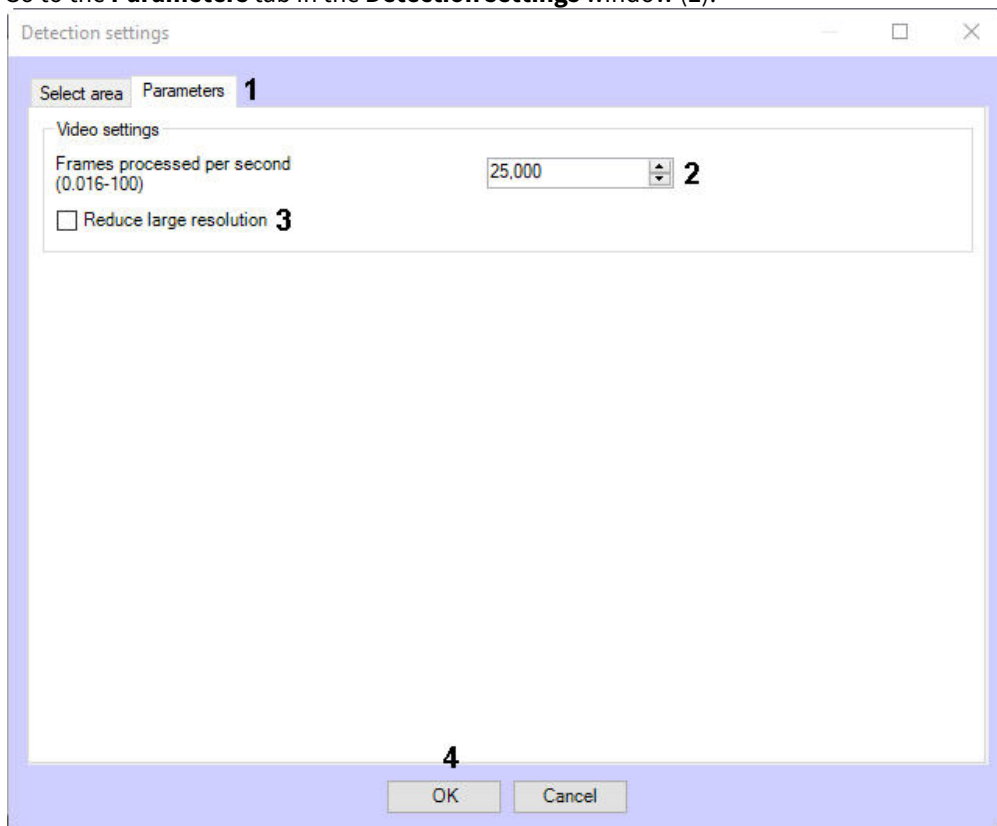
Note.

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. In the **Frames processed per second (0.016-100)** field (2), set the number of frames per second that will be processed by the detection tool.
- c. Set the **Reduce large resolution** checkbox to create and process the new frame consist of even lines of initial frame (3).

- Click **OK (4)**.

Configuring the *People counter detection* module is complete.

4.6 Stopped vehicle detection

4.6.1 Functionality of the Stopped vehicle detection module

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

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

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

4.6.3 Configuring the Stopped vehicle detection module

4.6.3.1 Licensing the Stopped vehicle detection module

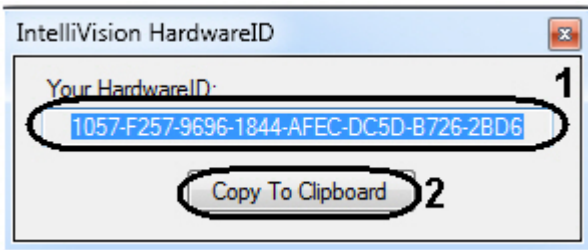
To license the «Stopped vehicle detection» module do the following:

- Go to the <Intellect installation folder>\Modules\IntelliVision folder.
- 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.

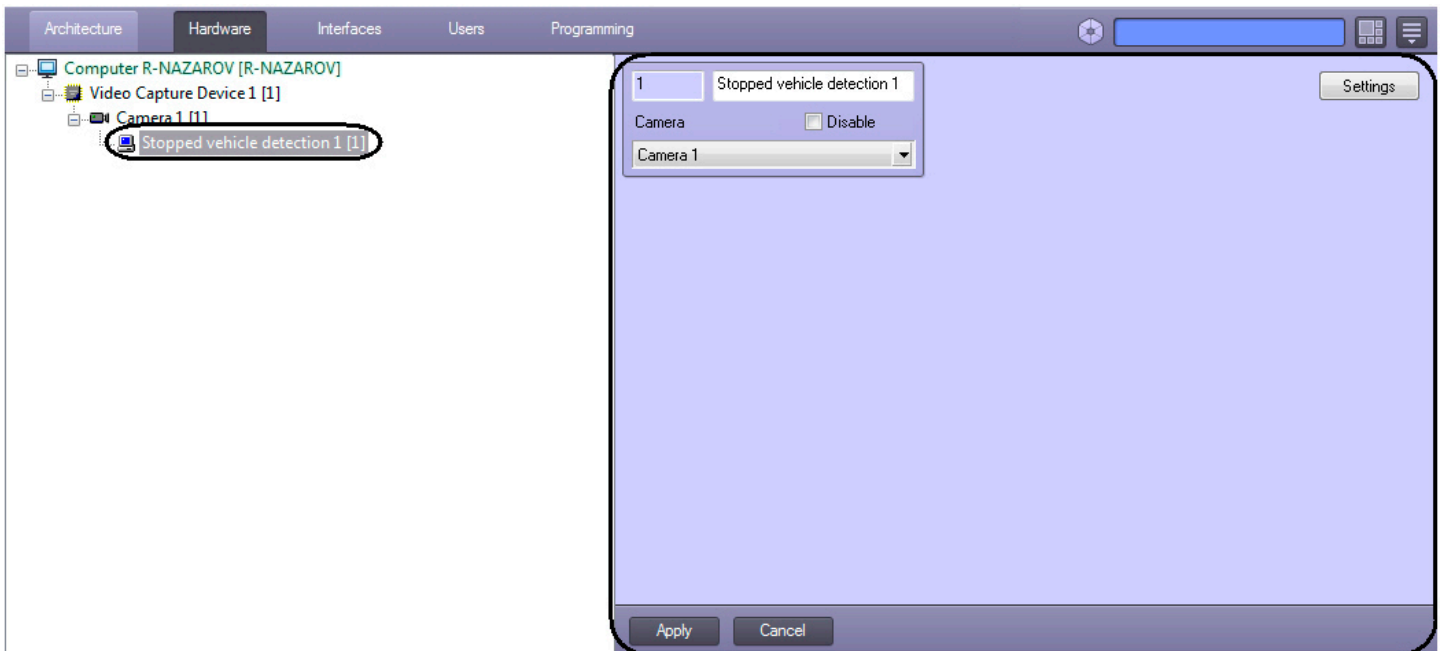


3. In the **Your HardwareID** field the code is displayed (1).
4. Click the **Copy To Clipboard** button to copy the code to the clipboard (2).
5. Send the code to the manager of the ITV company and specify the number of **Stopped vehicle detection** which are planned to be used.
6. Receive the regkey.dat file from the manager of the ITV company.
7. Put the received file to the <Intellect installation folder>\Modules64\ folder.

Licensing the «Stopped vehicle detection» module is completed.

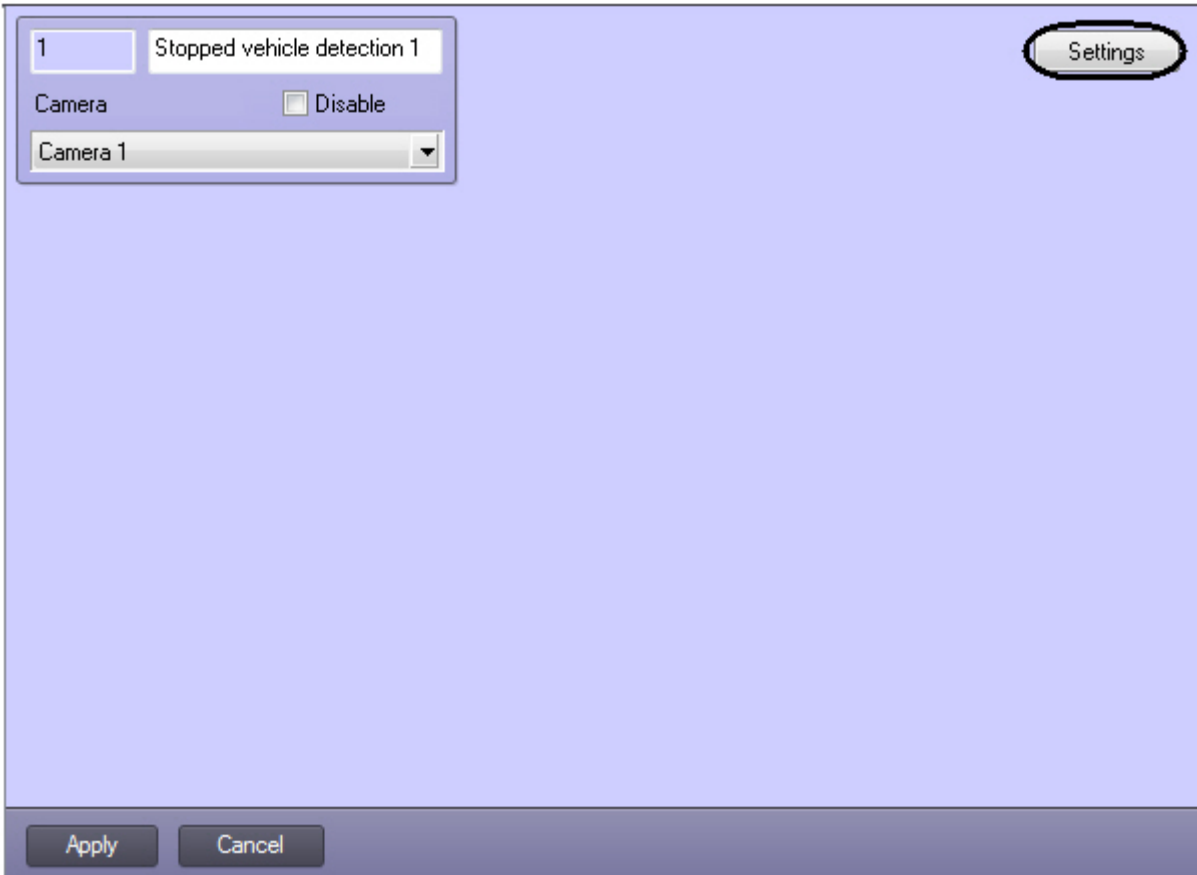
4.6.3.2 Configuring the Stopped vehicle detection module

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:

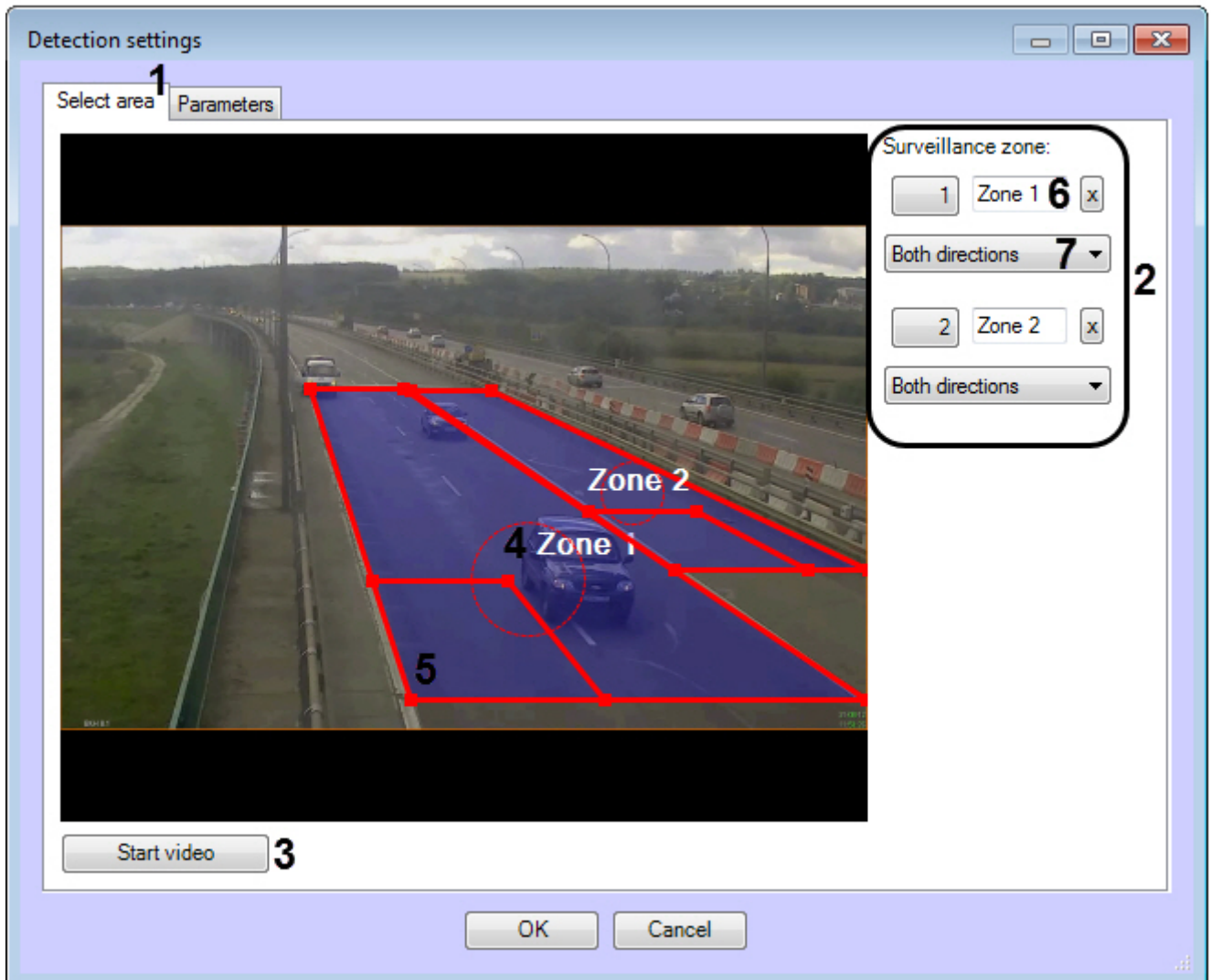
1. Go to the **Stopped vehicle detection** settings panel.



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.

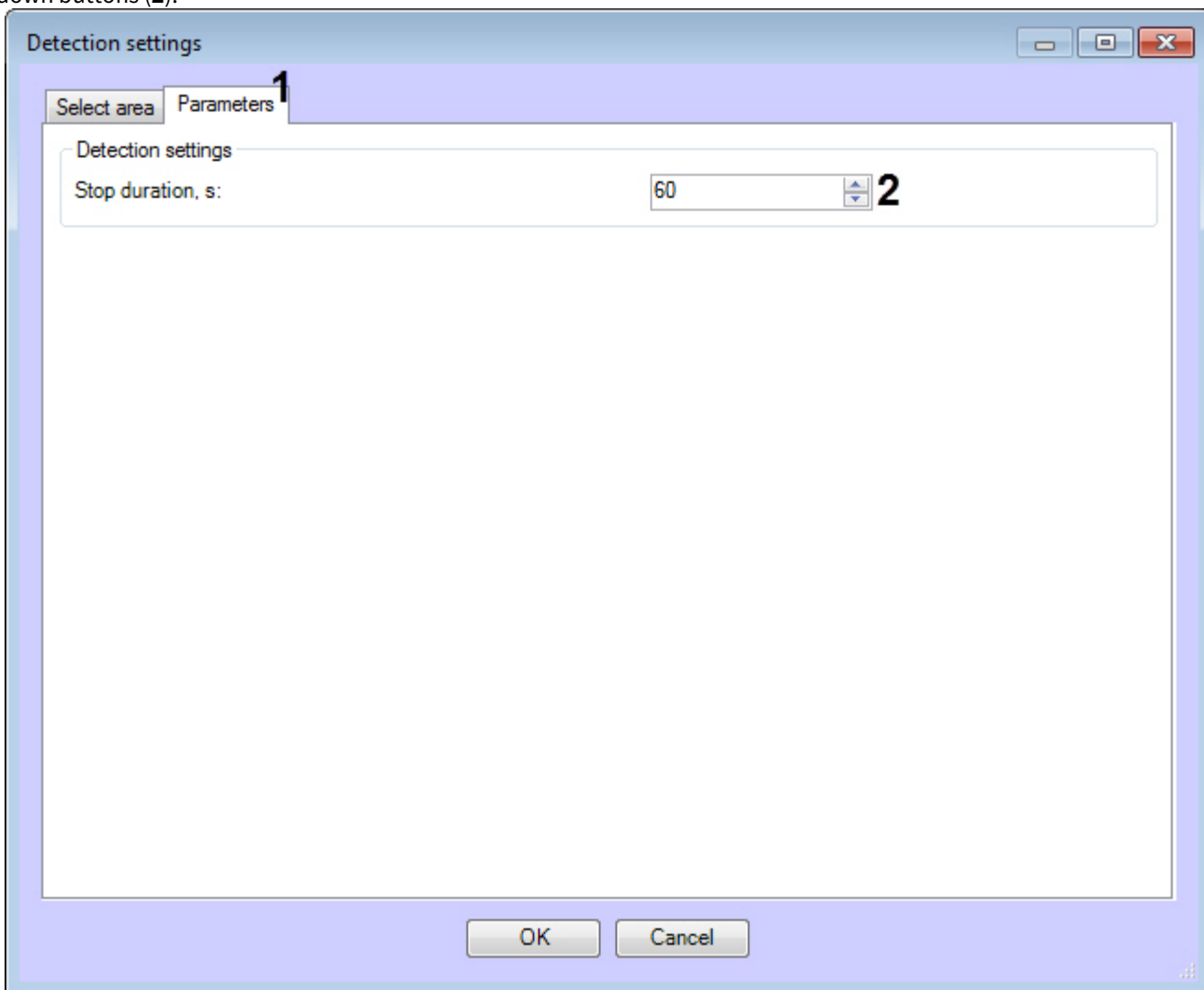
Note.

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.

4.7 Detection of light indication control

4.7.1 Functionality of the Detection of light indication control module

The **Detection of light indication control** 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.

4.7.2 Camera requirements for the Detection of light indication control module

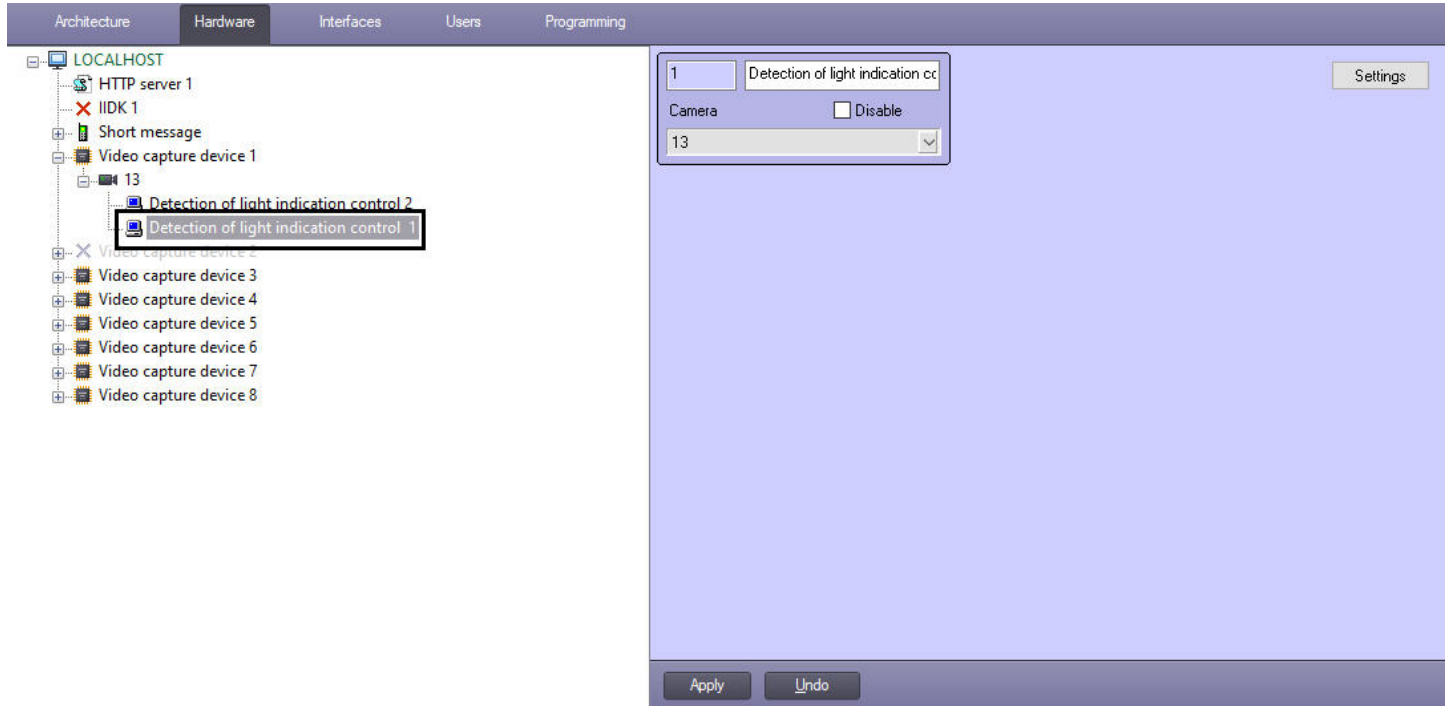
The requirements for the cameras that will work with the **Detection of light indication control** module are listed in the following table.

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

<p>Scene and camera angle</p>	<ul style="list-style-type: none"> 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).
<p>Objects image</p>	<ul style="list-style-type: none"> Light sources in the video image should be visually distinguishable.

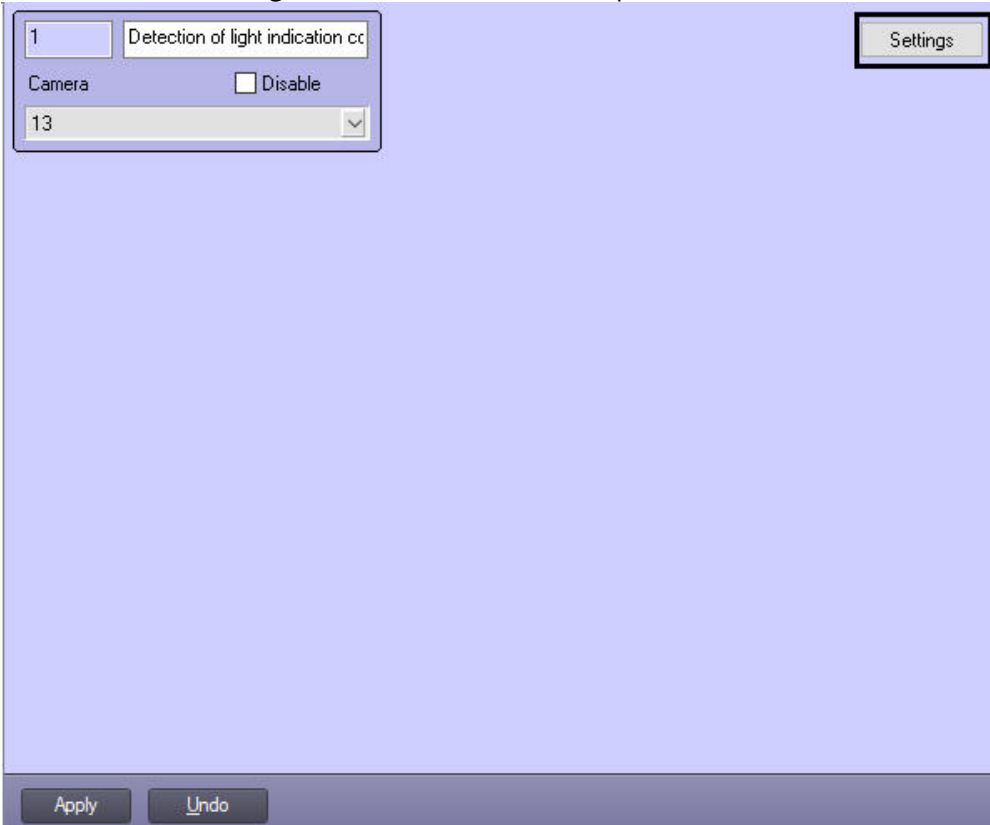
4.7.3 Configuring the Detection of light indication control module

The **Detection of light indication control** module is configured using the **System settings** menu, under the **Hardware** tab, on the **Detection of light indication control** control panel, using the **Camera** settings.

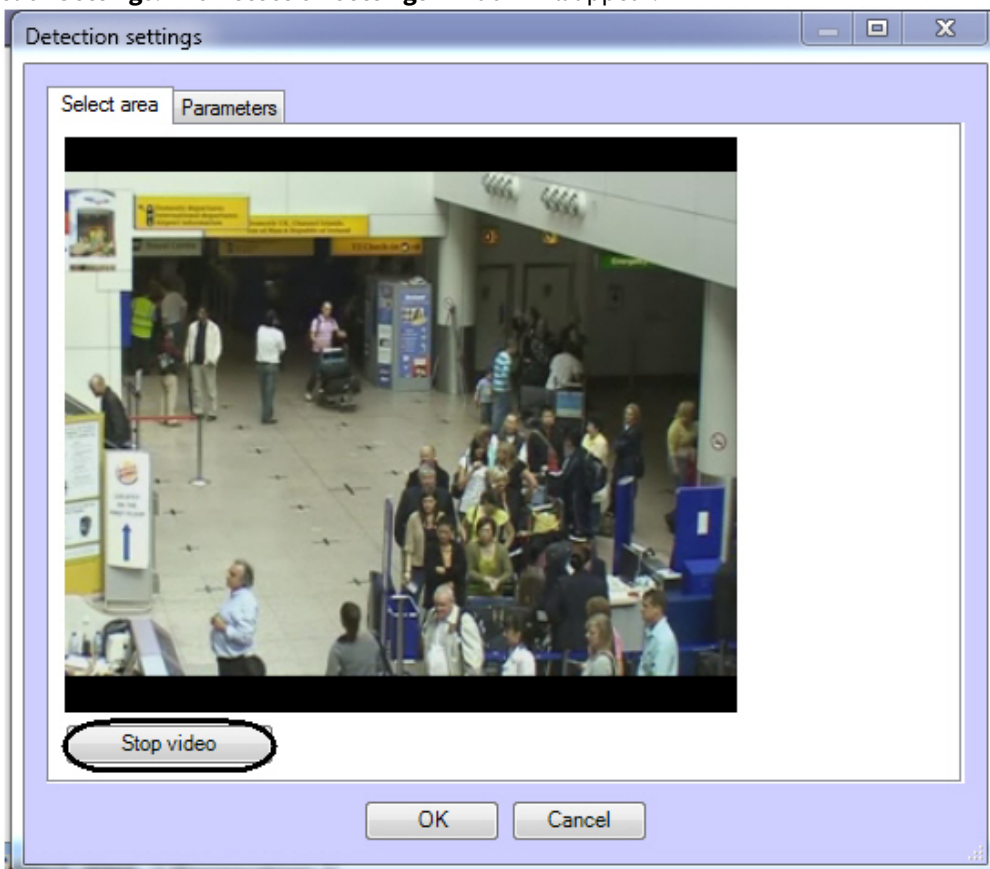


The **Detection of light indication control** module is set up as follows:

1. Go to the **Detection of light indication control** control panel.




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



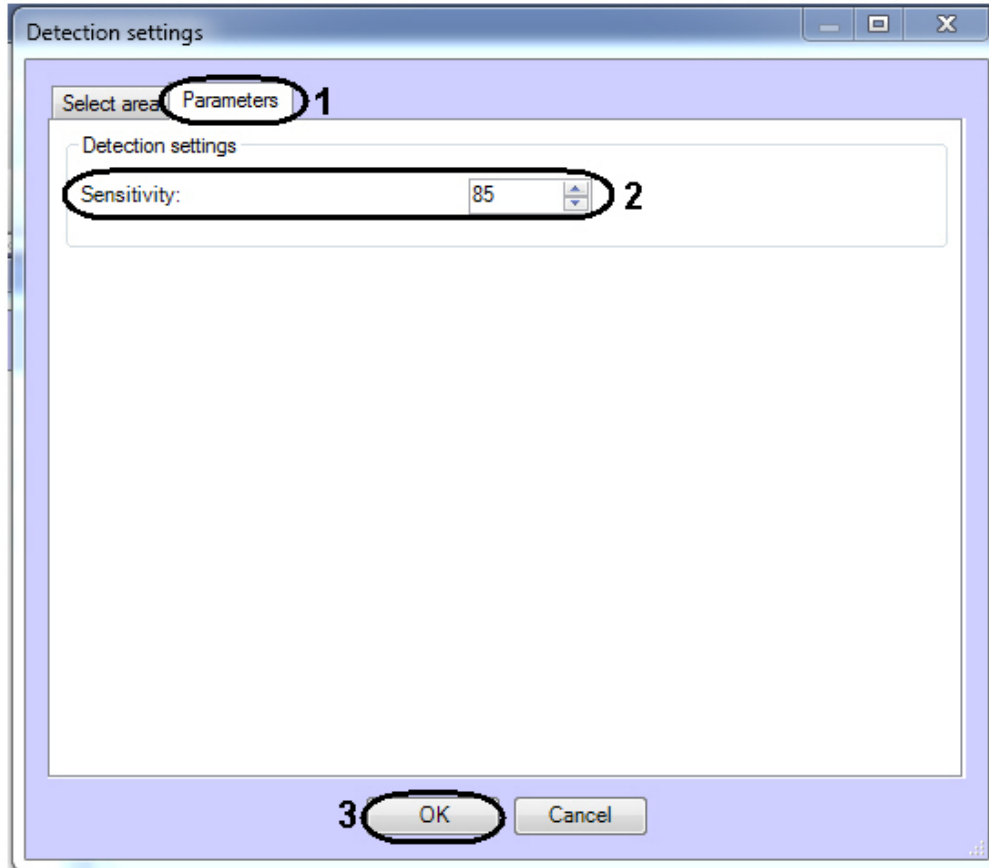
3. Specify the location of lights sources in the image which are to be tracked by detection:
 - a. Click **Stop video** to capture the video image.

- b. On the captured video image specify areas to be analyzed (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 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 **Detection of light indication control** module is complete.

4.8 Heat map detection

4.8.1 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.

4.8.2 Camera requirements for the Heat map detection module

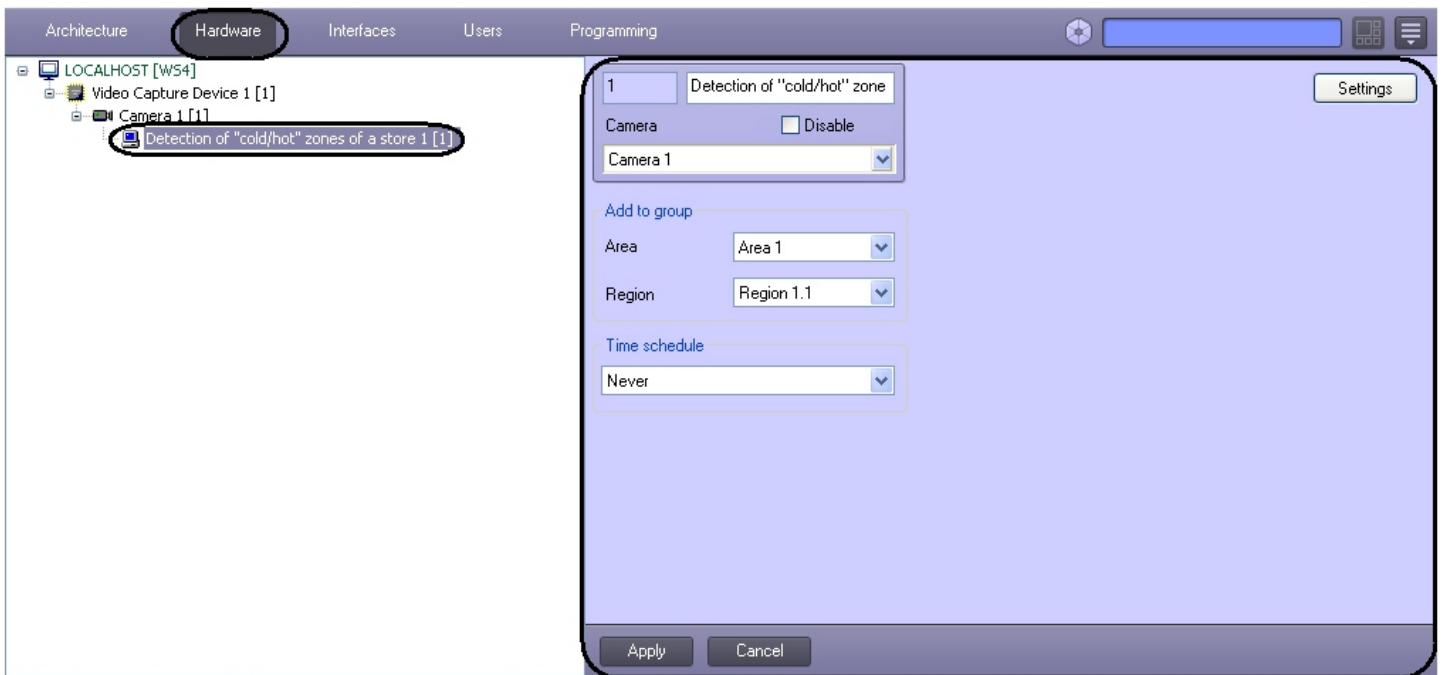
The requirements for the cameras that will work with the **Heat map detection** module are listed in the following table.

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

<p>Scene and camera angle</p>	<ul style="list-style-type: none"> • Sharp changes of lighting can lead to short-time invalid analytics working. • The best position – camera "looks" to the scene vertically down. The better this requirement, the carefully the received estimation. • Sizes of camera field of view: 3x3m is minimal (6x6 people), 4x4m is optimal (8x8 people), 8x8m is maximal (16x16 people). • Background is static and is not changed sharply. • Analytics can work inappropriately on specular surfaces and in case of sharp shadows from moved objects. • Analytics can work inappropriately in case of in the camera field of view there are periodic movements of background objects (trees, working TV, etc.)
<p>Objects image</p>	<ul style="list-style-type: none"> • Image quality: the image is to be clear, without visible defects from reducing procedure. • 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%

4.8.3 Configuring the Heat map detection module

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:

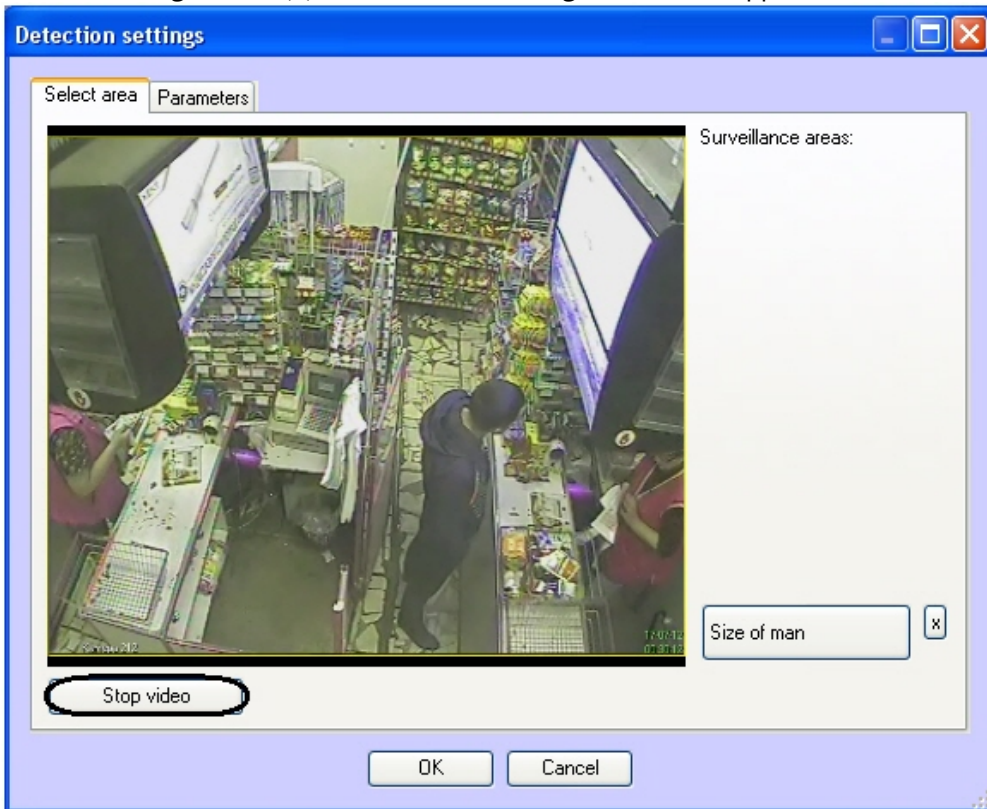
1. Go to the **Heat map detection** control panel.

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!

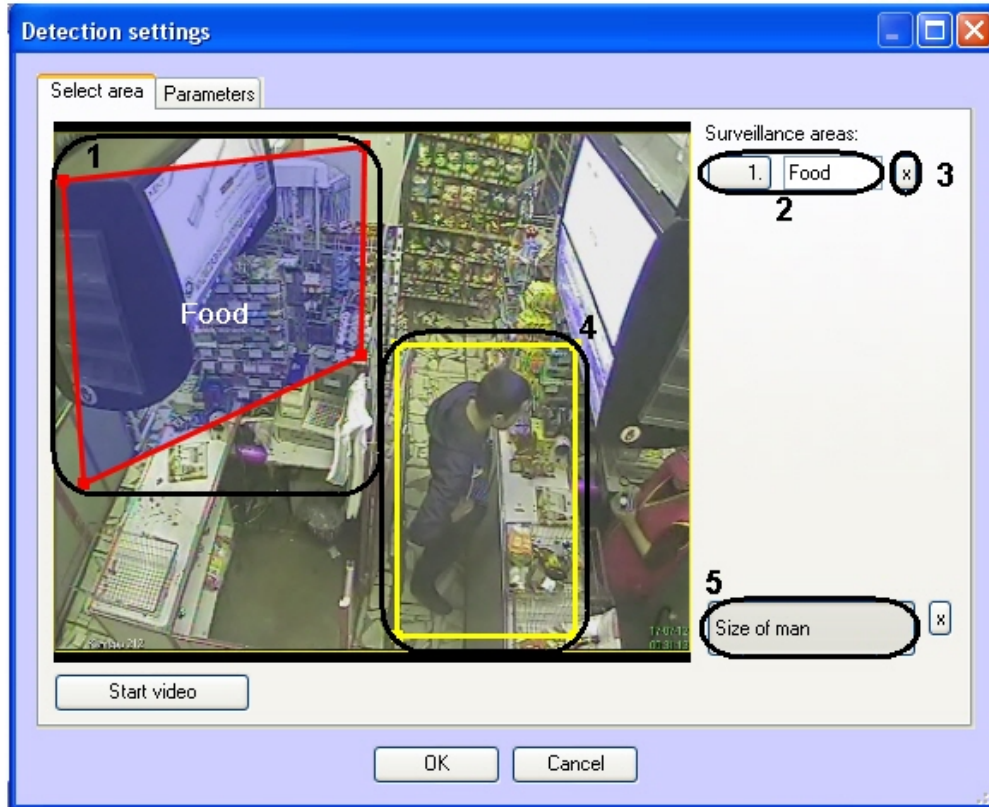
If no custom **Time zone** objects is created in the system, two time zones are available – **Always** and **Never**. The **Never** time zone is set by default, so the detection tool will not work if default settings are left unchanged.

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



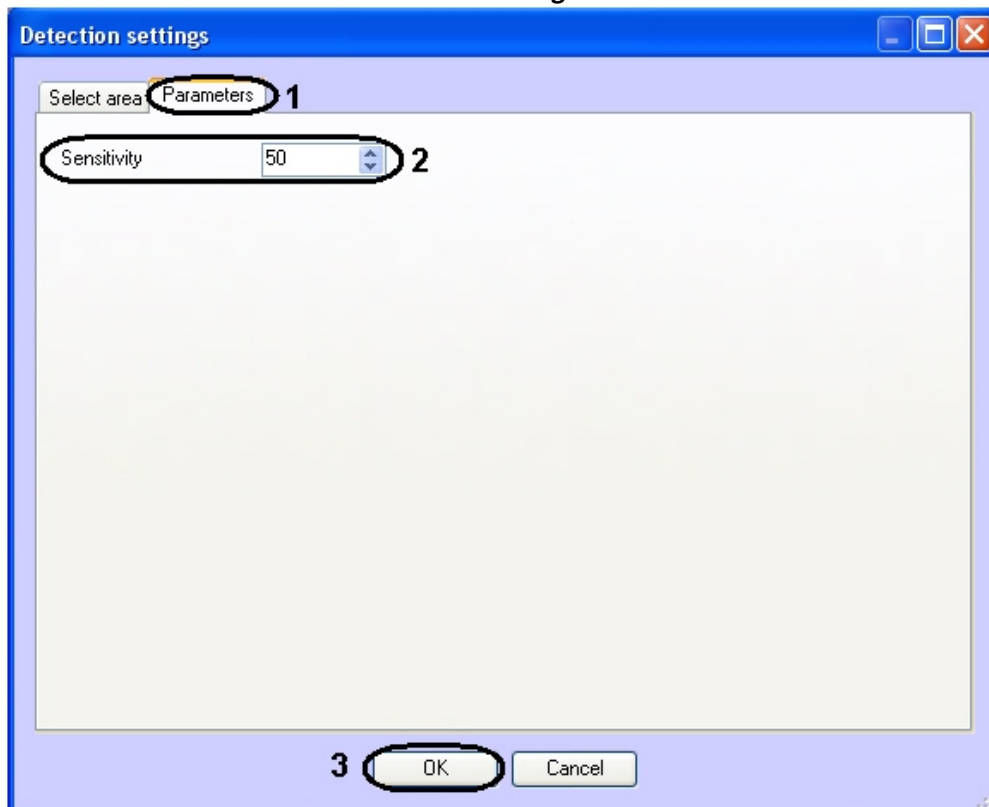
6. Specify the area of interest and the approximate size of people in the video image:
- Click the **Stop video** button to capture the video image.
 - On the captured video image specify areas to be analyzed (1). To specify the area set the nodal points of interested area using the left mouse button. 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.

4.9 Detection of moving against crowd flow

4.9.1 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.

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

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

4.9.3 Configuring the Detection of moving against crowd flow module

4.9.3.1 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.

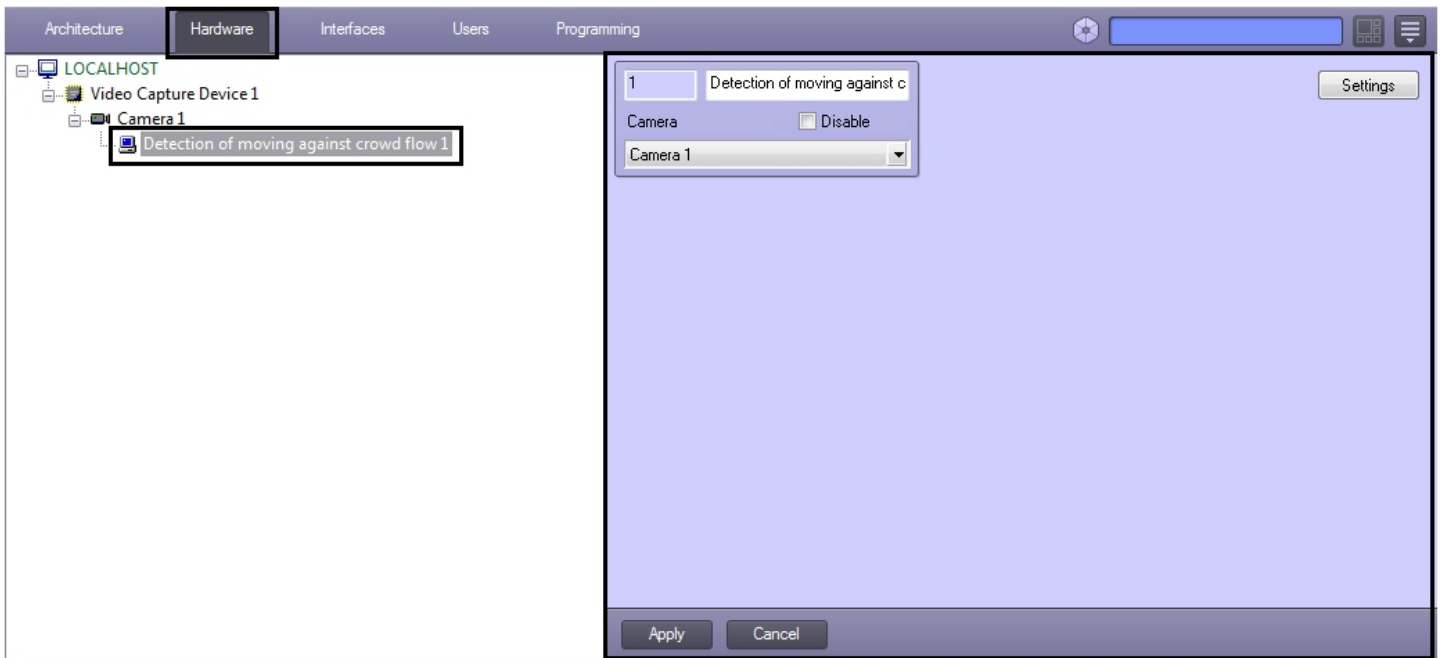
There will not new detection triggers while a person on which detection functioned is stayed in the monitored area.

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.

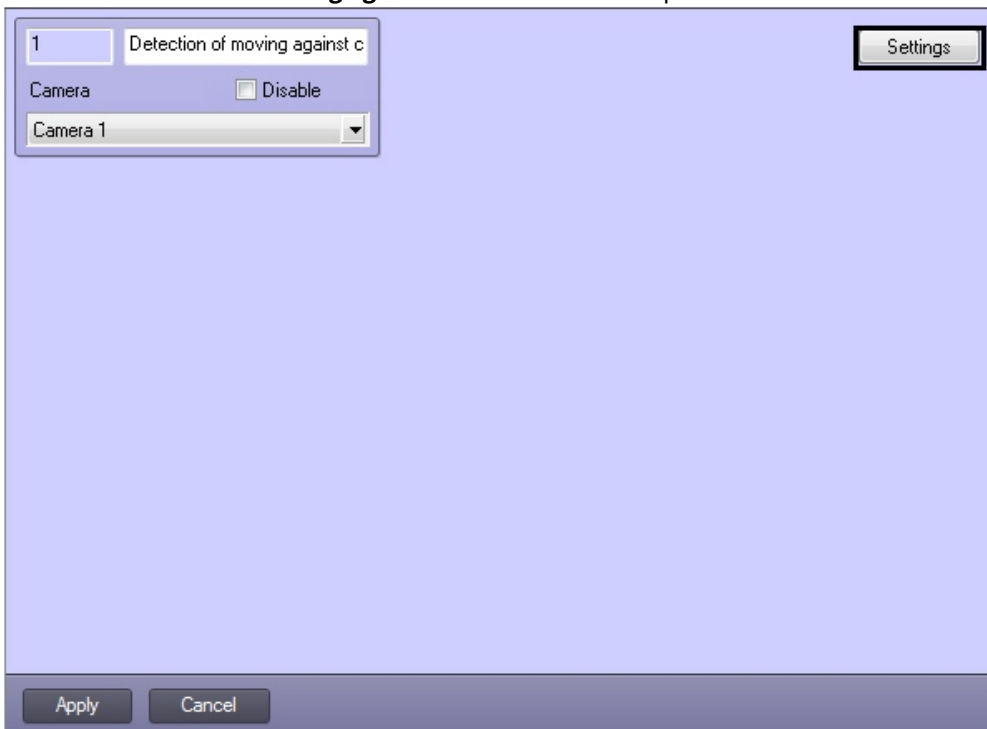
4.9.3.2 Setting up the Detection of moving against crowd flow module

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.

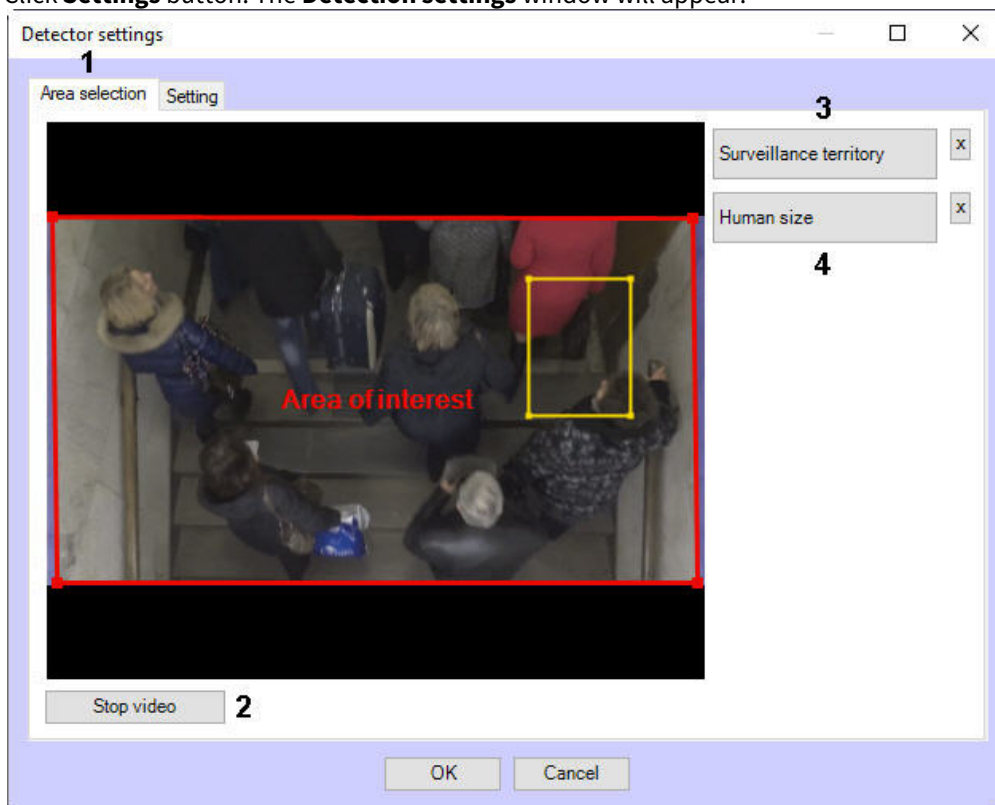


The *Detection of moving against crowd flow* module is set up as follows:

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 surveillance area:
- Go to the **Select area** tab (1).
 - Click the **Stop video** button to capture the video image (2).
 - Click the **Area of interest** button (3).
 - 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.

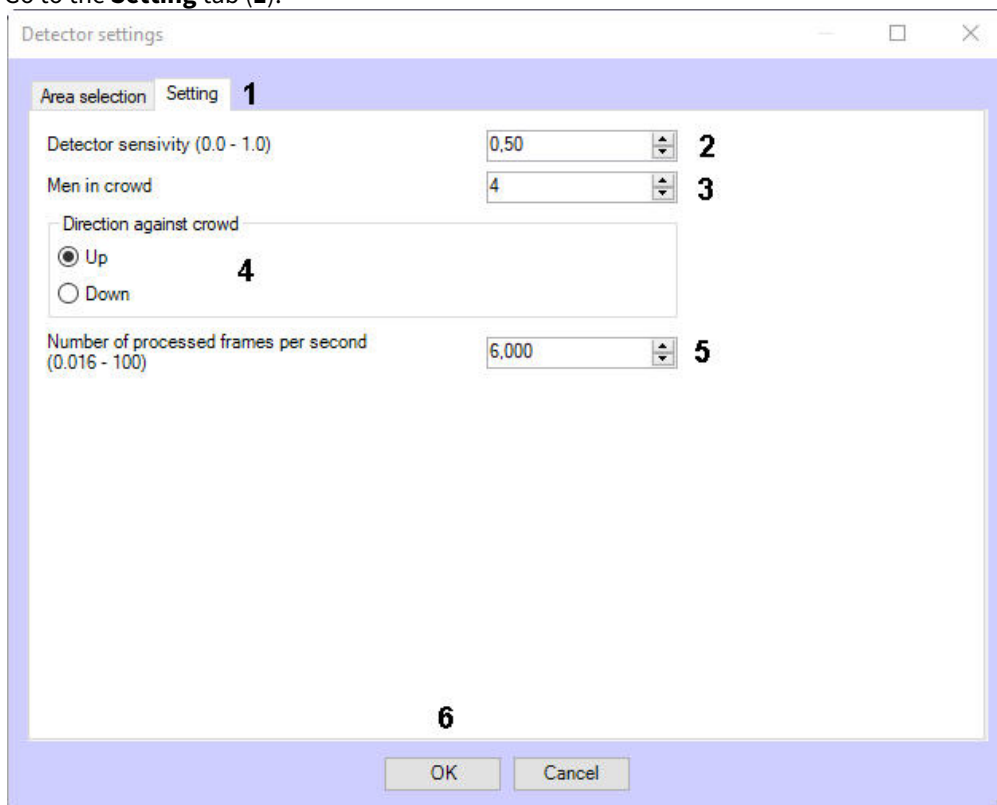
- Click the **Size of man** button (4). Set the required person size. To do this, click the left mouse button on the captured video image and extend a rectangular area to required size.

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 **Setting** 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).
- e. In the **Number of processed frames per second (0.016 - 100)** field (5), set the number of frames per second that will be processed by the detection tool. The higher the value, the more accurate the detection tool works, but the higher the CPU load.
5. Click the **OK** button (6).

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

4.10 Sweethearting at checkout detection

4.10.1 Functionality of the Sweethearting at checkout detection module

The **Sweethearting at checkout 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 at checkout 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 at checkout 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.

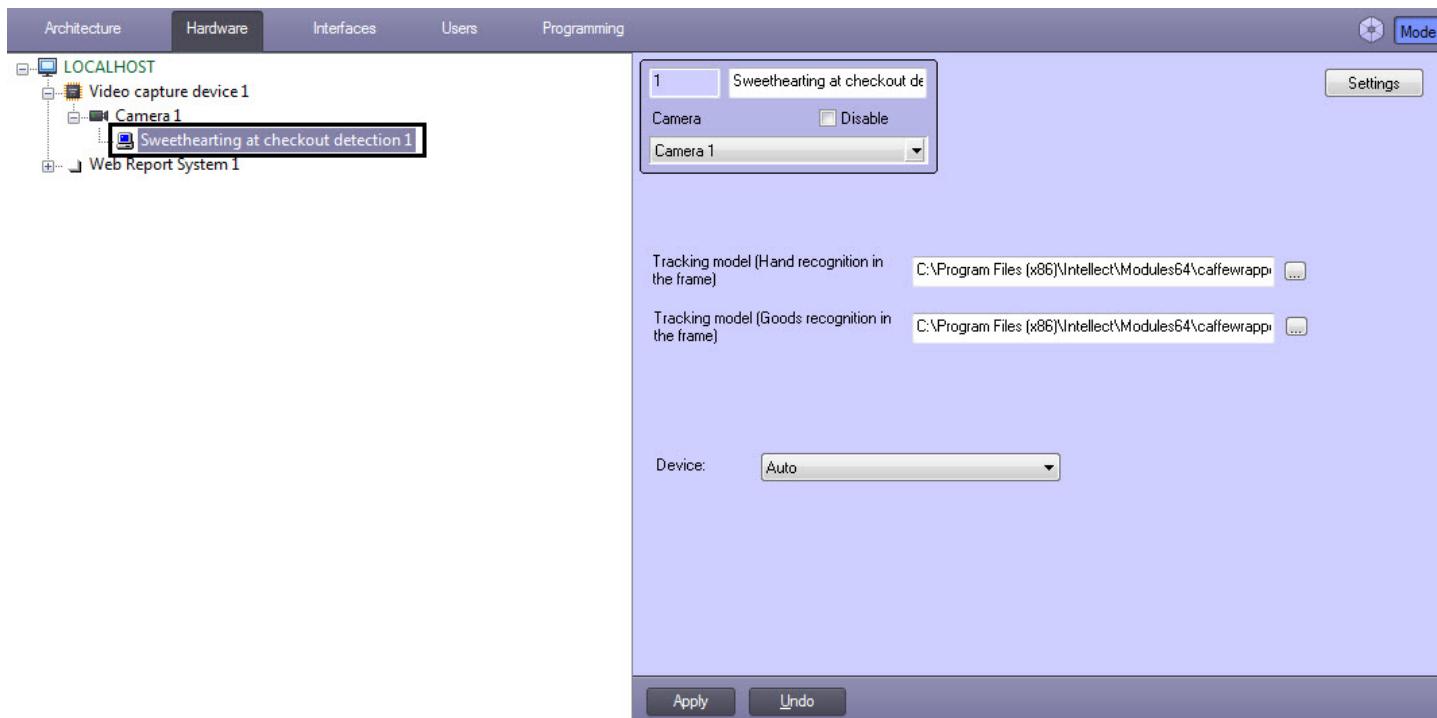
4.10.2 Camera requirements for the Sweethearting detection module

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

Camera	<ul style="list-style-type: none"> Resolution should be at least 1920x1080 pixels Fps not less than 12 Only color cameras
Scene and camera angle	<ul style="list-style-type: none"> 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.
Object images	<ul style="list-style-type: none"> The area of interest of the detection tool (see Configuring the «Sweethearting detection» module) 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).
Lighting requirements	<ul style="list-style-type: none"> Objects should be visually separated from the background and from each other

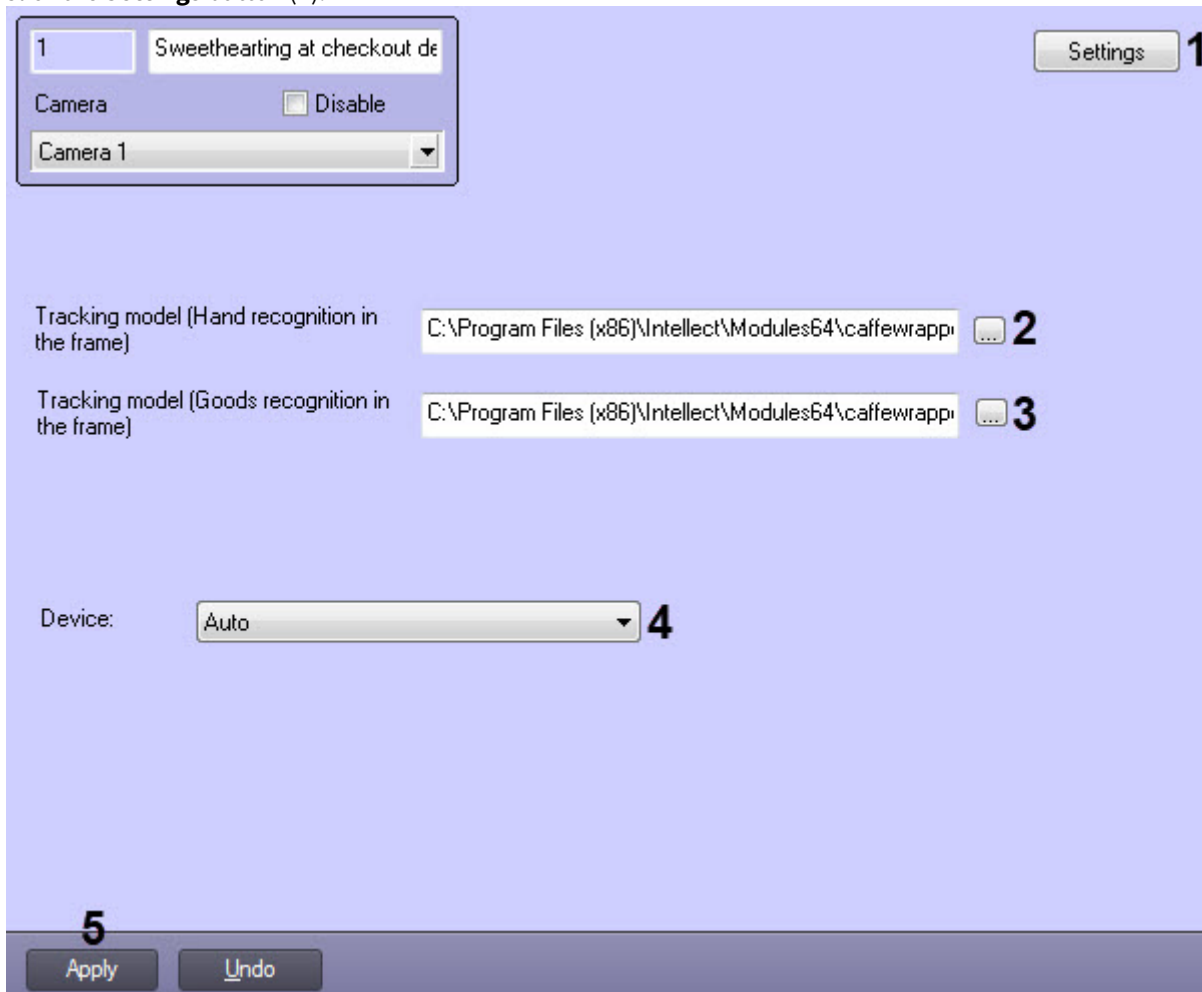
4.10.3 Configuring the Sweethearting at checkout detection module

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

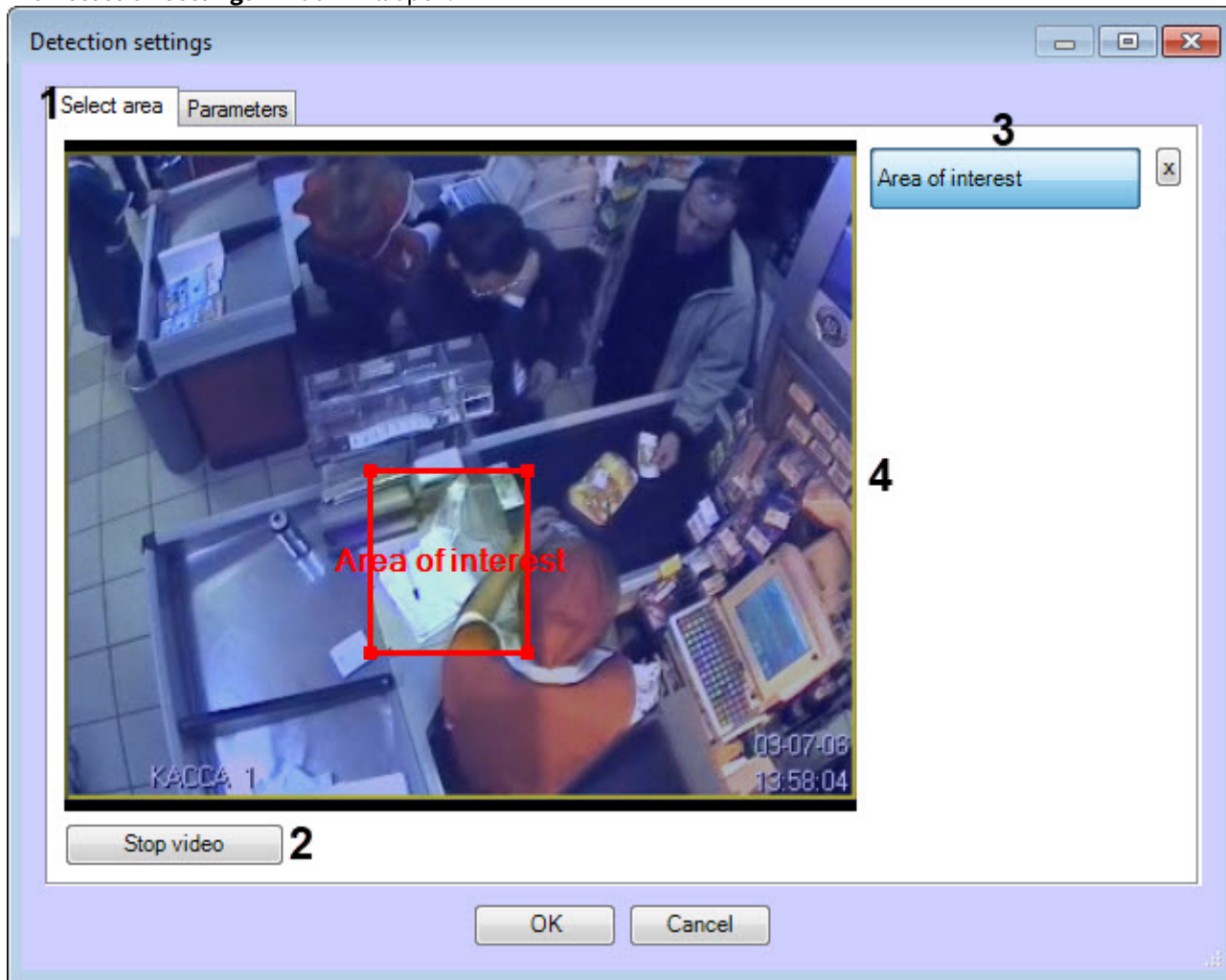


The **Sweethearing at checkout detection** module is configured as follows:

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




The **Detection settings** window will open.

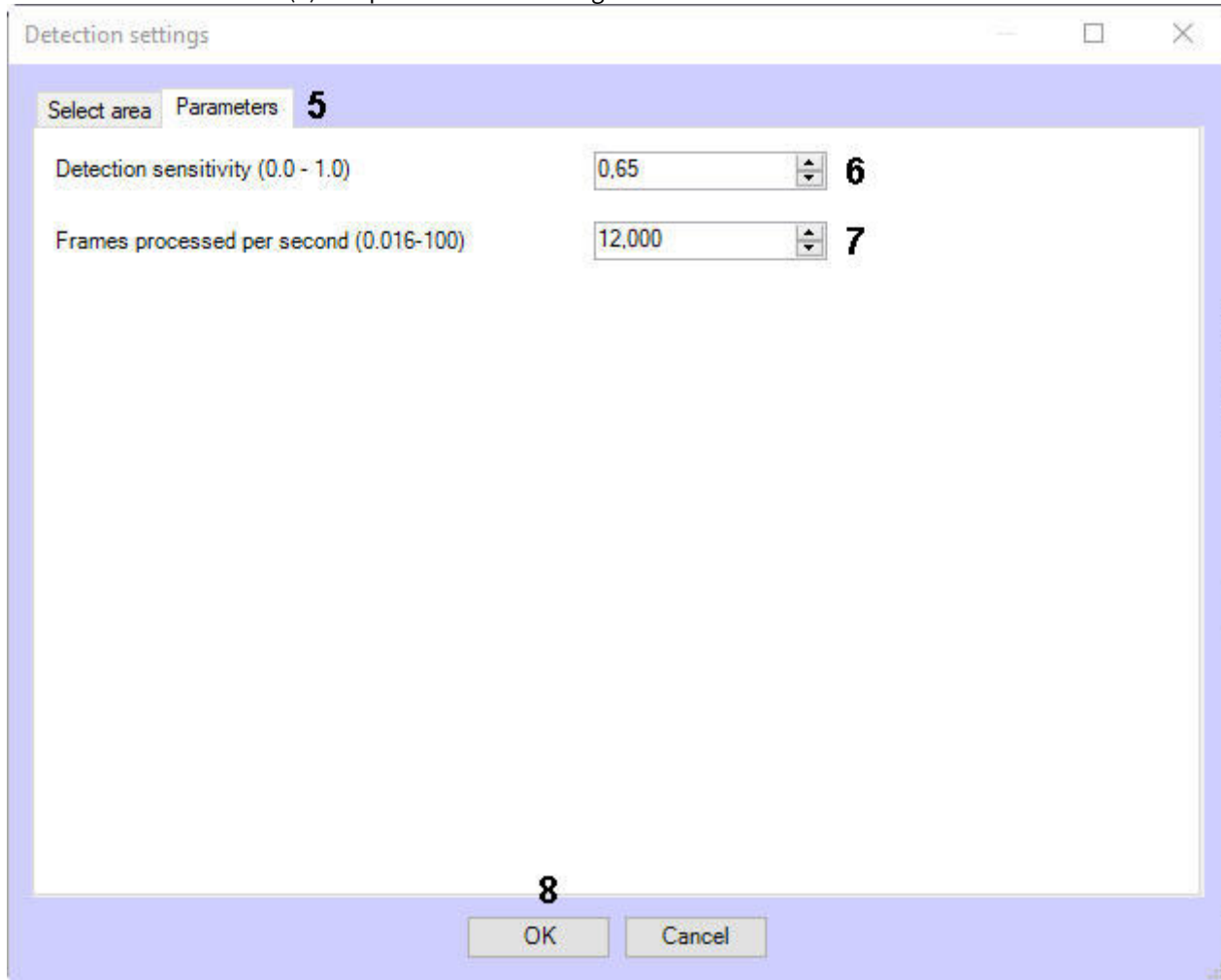


- a. Specify the area of interest of the detection:
 - i. Go to the **Select area** tab (1).
 - ii. Click **Stop video** (2) to capture the video image.
 - iii. Click **Area of interest** (3).
 - iv. Specify the area of interest in the captured video image to be analyzed (4). The selected area must comply with [Camera and camera settings requirements for the Sweethearting 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.

b. Go to the **Parameters** tab (5) and perform the following actions:



c. Specify the detection sensitivity in the range from 0.0 to 1.0 in the **Detection sensitivity (0.0 - 1.0)** parameter (6).

Note

The detection sensitivity value is selected experimentally. The lower the sensitivity, the greater the probability of false positives. The higher the sensitivity, the less chance of false alarms, however, some useful tracks may be skipped.

- d. In the **Frames processed per second (0.016-100)** field (7), set the number of frames per second that will be processed by the detection tool. Default value is 12.
- e. Click **OK** (8) to save changes and return to the settings panel of the **Sweethearing at checkout detection** object.

Note

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

3. If you use a unique neural network, select a neural network file with the hand recognition in the frame (2) and goods recognition in the frame (3) tracking model. It is not necessary to select standard neural networks in this field, the system will automatically select the required one. Standard neural networks are located in the *C:\Program Files (x86)\Intellect\Modules64\caffewrapper\Networks* directory:

dpe_224_2cl_hands_v6_50k.ann	Neural network file with the hand recognition in the frame tracking model
dpe_224_2cl_product_v6_122_5k.ann	Neural network file with the goods recognition in the frame tracking model

4. In the **Device** drop-down list (4) select the device where the neural network will operate.

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

The **Sweethearting at checkout detection** module is now configured.

4.11 Barcode detection

4.11.1 Functionality of the Barcode detection

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

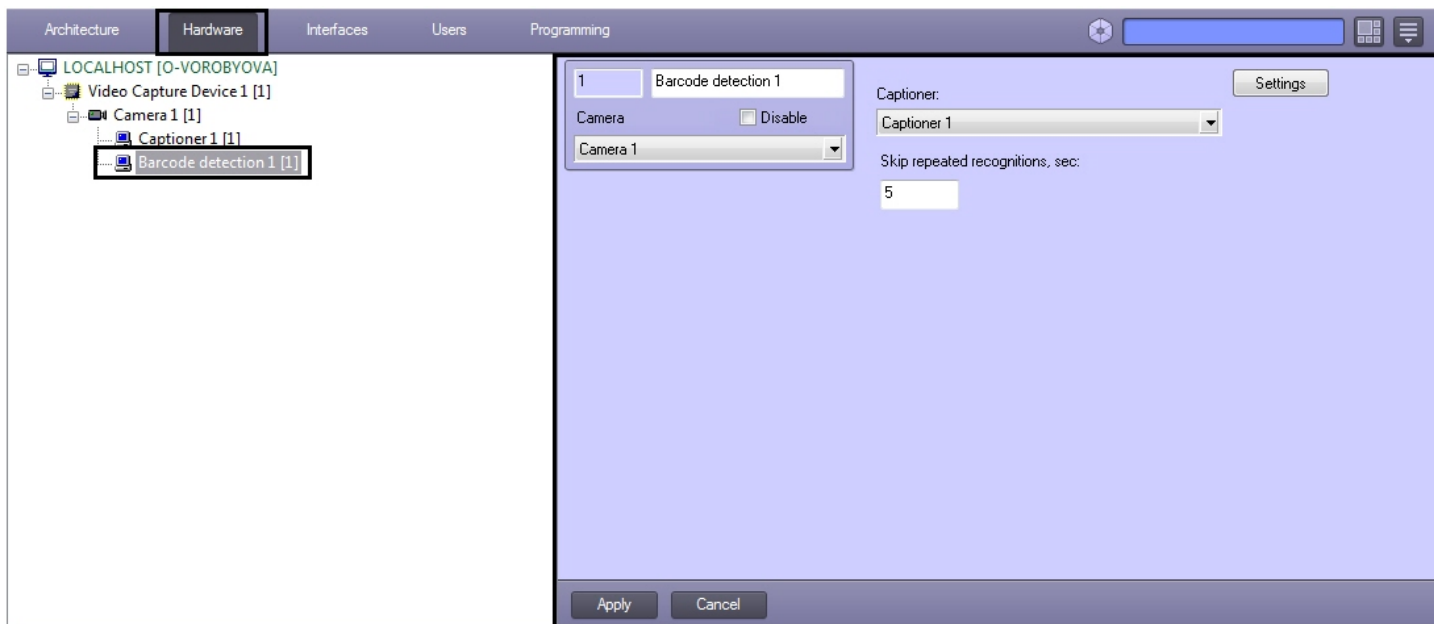
4.11.2 Camera requirements for the Barcode detection module

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"> 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. FPS rate: 6
Lighting	<ul style="list-style-type: none"> The image should be clear, lines should be visually separable from each other
Scene and camera view	<ul style="list-style-type: none"> The angle between the optical axis of the camera and the plane of the barcode should be not more than 15°
Images of objects	<ul style="list-style-type: none"> Maximum width and height of the detected barcode – 65536 pixels The area of the detected barcode is not less than 1296 pixels Each of barcode sides is not less than 10 pixels <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>Note. 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>

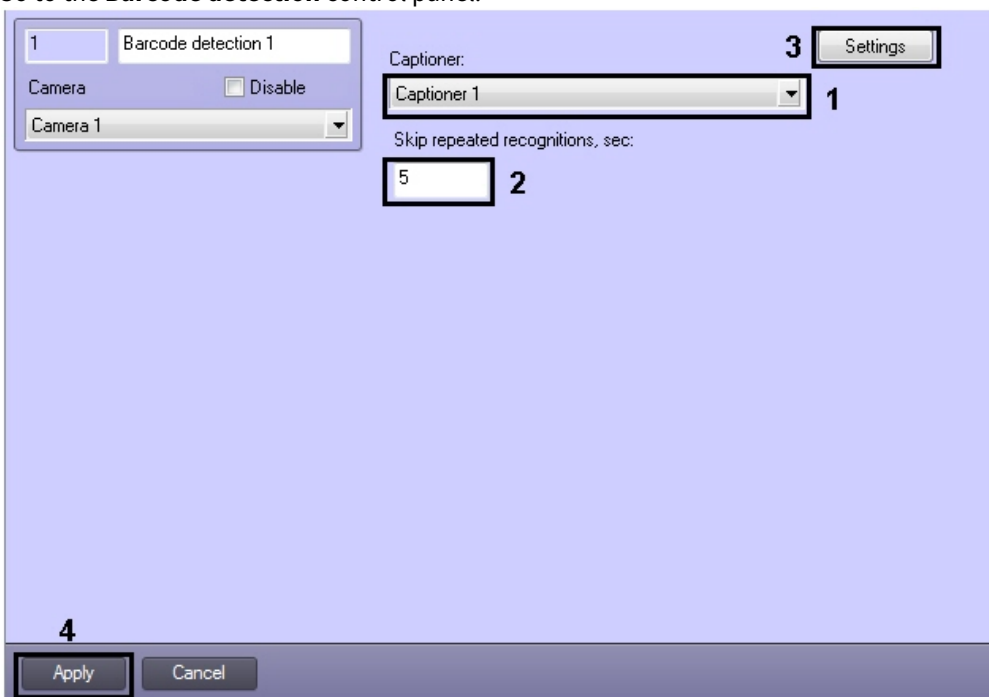
4.11.3 Configuring the Barcode detection module

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.

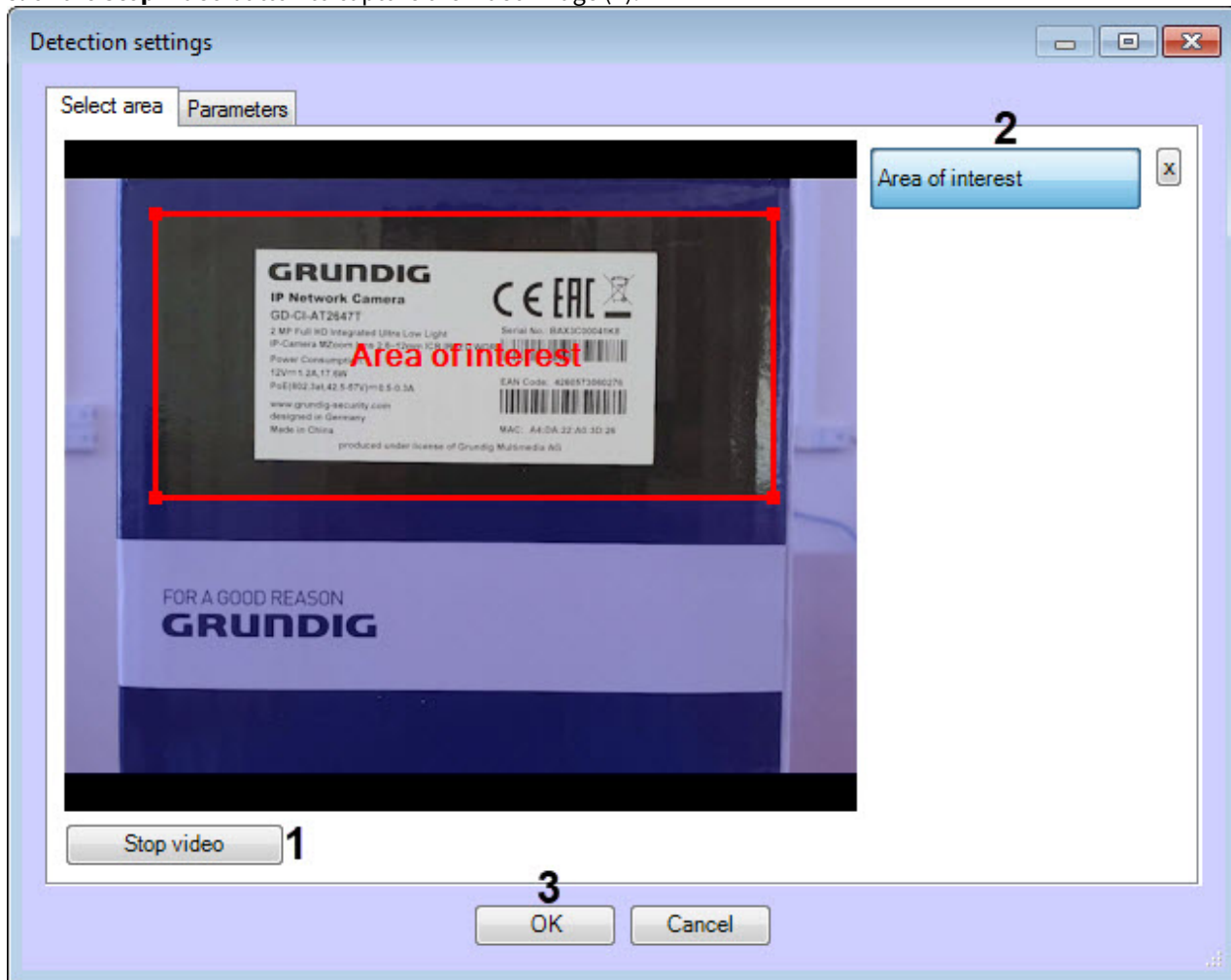


2. From the **Captioner** drop-down list select the captioner with the help of which result will be displayed in the monitor (1)
3. In the **Skip repeated recognitions, sec** field enter the time in seconds in which the repeated code is recognized (2).


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. 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 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.

Attention!
Barcode can be detected not only when the entire barcode or QR code is within the specified area, but also only a fragment of it.

- d. Click **OK** button (3) to save changes and return to the control panel of the **Barcode detection**.

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** button (4).

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

4.12 Train detection

4.12.1 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.

4.12.2 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.

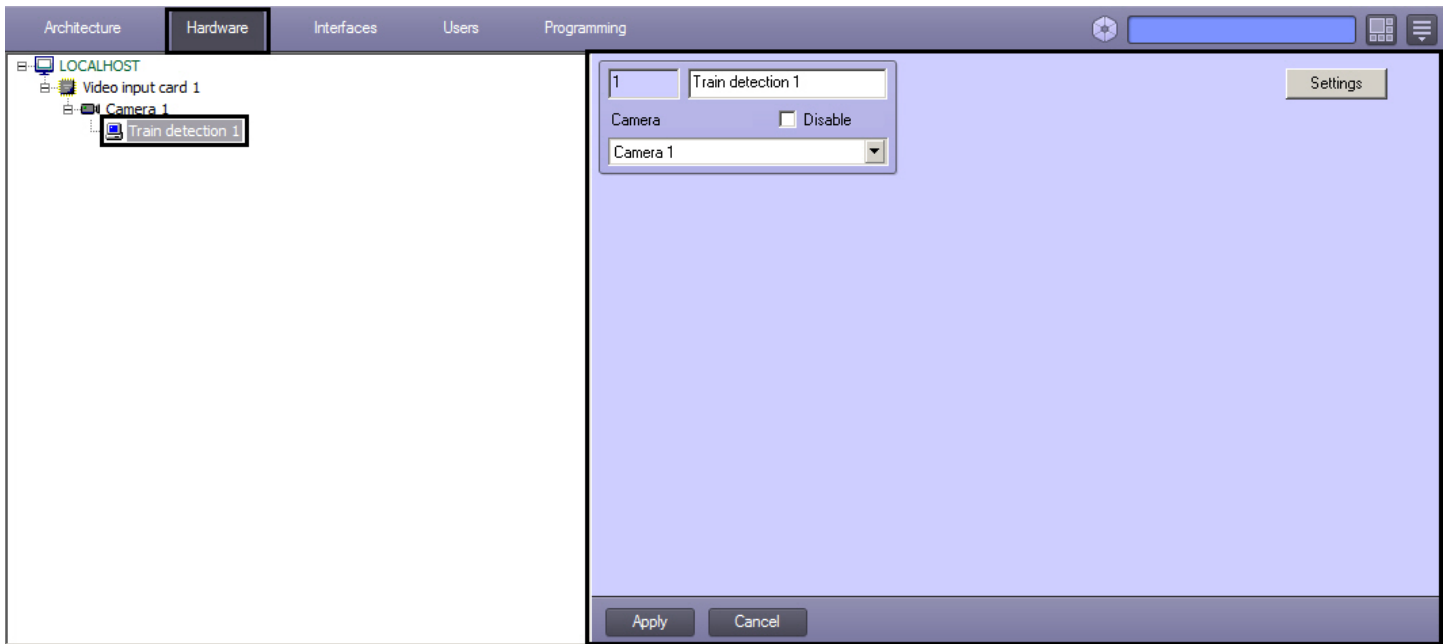
Camera	<ul style="list-style-type: none"> • Resolution should be at least 640x480 pixels. • Fps: not less than 6 • Color: analytics works with grey and color images. • Camera shaking should not result in image offsets greater than 1% of the frame size.
Lighting	<ul style="list-style-type: none"> • Medium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

4.12.3 Configuring the Train detection module

Note.

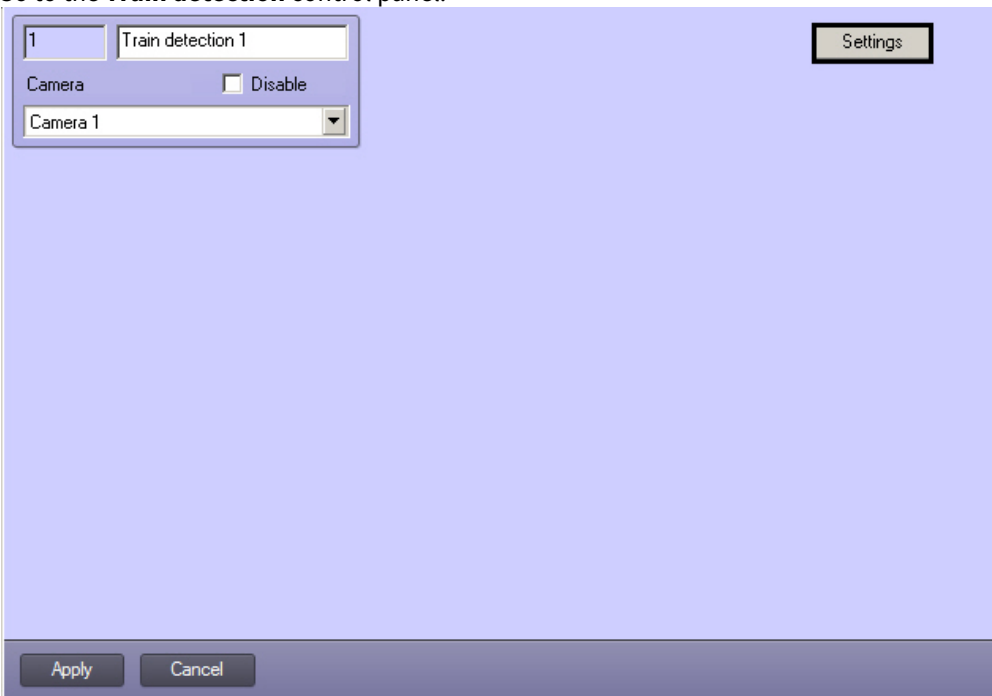
It is not recommended to create more than 4 **Train detection** objects for correct operation of the *Train detection* module.

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.

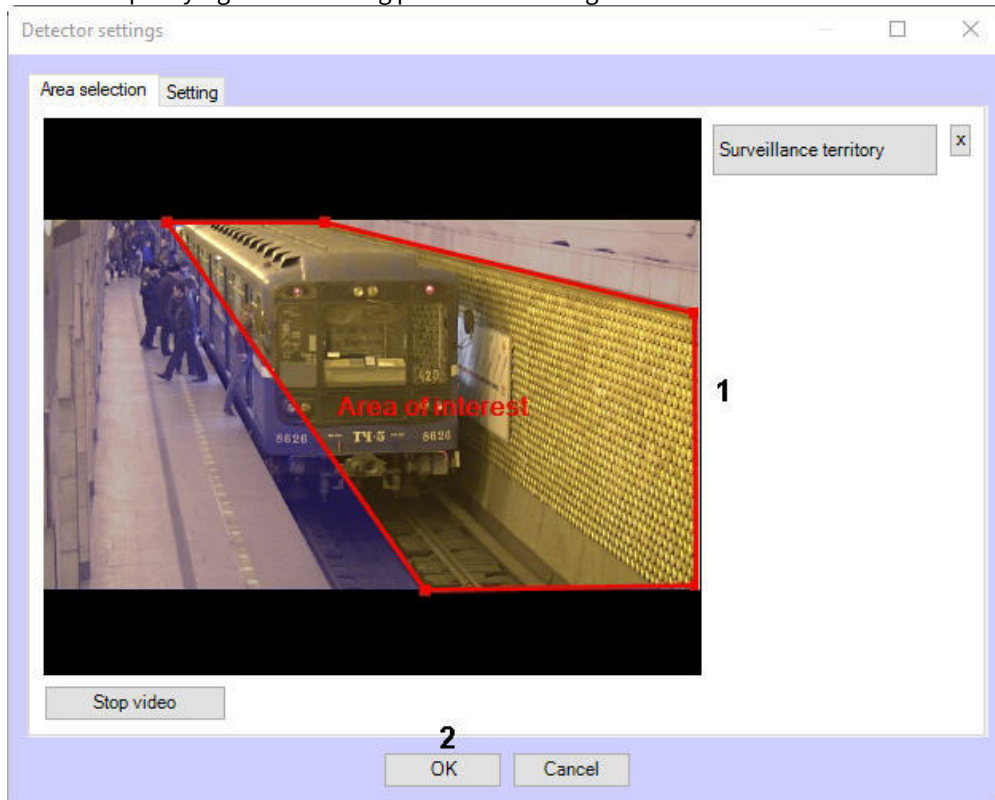


2. Click the **Settings** button. The **Detection settings** windows will open.
3. Specify the surveillance area on the video image:


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



- b. Click the **Surveillance territory** button (**2**).
- c. On the captured video image (**3**) sequentially specify nodal points of area to be analyzed 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 **Surveillance territory** button.

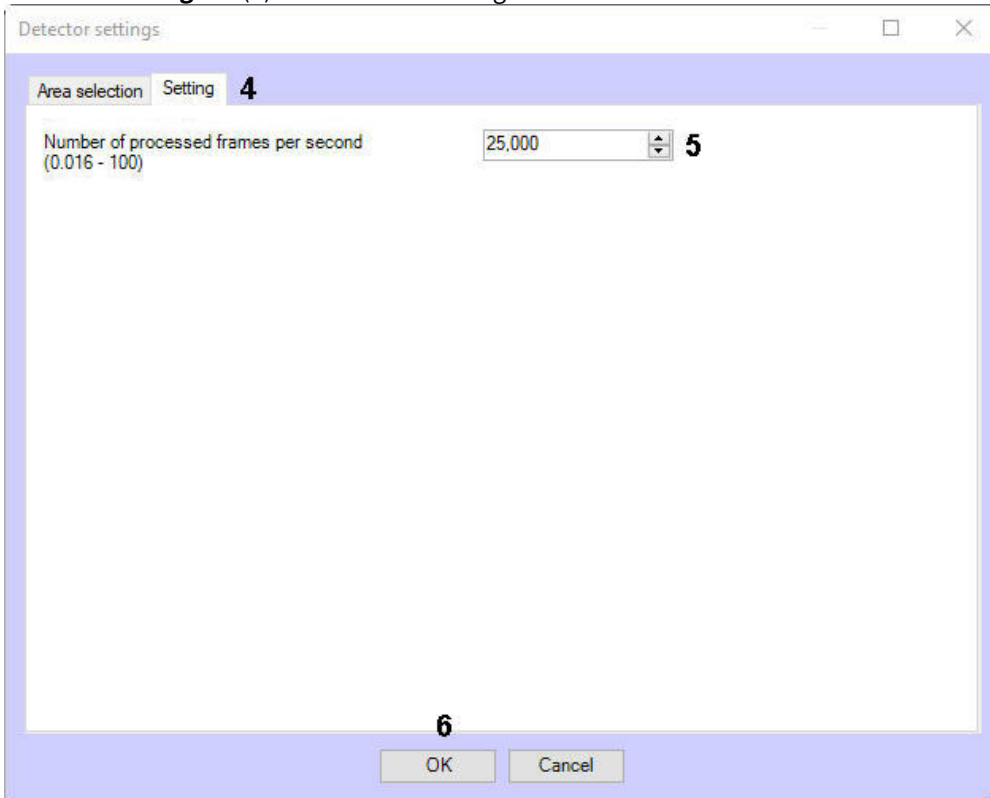
Note.

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

Attention!

Setting the **Surveillance territory** is a mandatory requirement for the detection module operation.

4. Go to the **Setting** tab (4) and do the following:



- a. In the **Number of processed frames per second (0.016-100)** field (5), set the number of frames per second that will be processed by the detection tool.
- b. Click the **OK** button to save changes and return to the control panel of the **Train detection** (6).

Note.

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

5. Click **Apply** on the **Train detection** control panel.

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

4.13 Fire detection and Smoke detection

4.13.1 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.

4.13.2 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"> • It is recommended to use color cameras. With black and white cameras, the recognition quality may be much worse • Resolution should be at least 640x360 pixels • 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)
Lighting	<ul style="list-style-type: none"> • Fire/smoke should be visually separated from the background
Object images	<ul style="list-style-type: none"> • The minimum area of fire/smoke in the frame depends on the neural network used. For a standard neural network (see Configuring the Fire detection and Smoke detection modules), the area of fire/smoke should be at least 10% of the frame. In some cases, when the fire is well contained, it may be sufficient for a detection tool that the width/height of the fire zone is 1-3% of the width/height of the frame.

Attention!

If the area of interest is specified on the video image, then the above requirements are relevant for it, and not for the entire frame (see [Configuring the Fire detection and Smoke detection modules](#))

4.13.3 Configuring the Fire detection and Smoke detection modules

Camera requirements for the Fire detection and Smoke detection modules

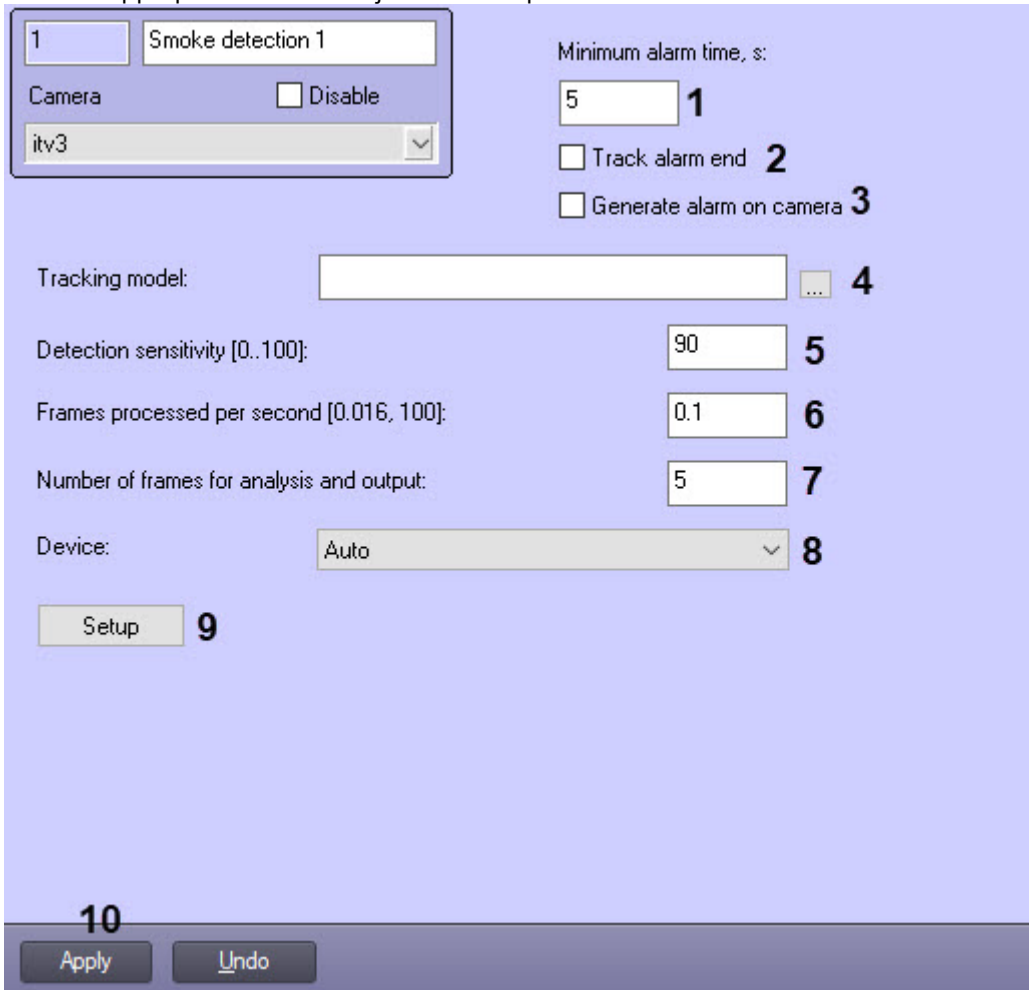
- The **Fire detection** module can be configured on the settings panel of the **Fire detection** object created on the basis of the **Camera** object on the **Hardware** tab of the **System settings** dialog window.

- 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:


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



- In the **Minimum alarm time, 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 **Minimum alarm time, 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. Set the **Track alarm end** checkbox (2) if it is necessary to generate an end-of-alarm event only after the expiration of the time, which is considered as: "time of the last received alarm" + "time specified in the **Minimum alarm time, sec** parameter".
4. Set the **Generate alarm on camera** checkbox (3) if it is necessary to initiate an alarm on the camera when a detection tool is triggered.
5. If you use a custom neural network, click the  button (4), and in the standard Windows box that opens, select the neural network file with the fire/smoke detection model. It is not necessary to select standard neural networks in this field, the system will automatically select the required one. Standard neural networks for different processor types used are located in the *C:\Program Files (x86)\Intellect\Modules64* directory:

smoke_openvino.ann	Smoke detection / CPU
smoke_original.ann	Smoke detection / GPU
fire_openvino.ann	Fire detection / CPU
fire_original.ann	Fire detection / GPU

6. In the **Detection sensitivity: 0..100** field (5) 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.

7. In the **Frames processed per second [0,016,100]** field (6), set the number of frames per second that will be processed by the detection tool. This parameter has an impact on processor capacity: the greater the parameter value, the more capacity. Also it has an impact on operation speed of algorithm: the greater the frame rate, the faster required number of frames will be collected for decision and alarm, if it's required.
8. In the **Number of frames for analysis and output** field (7) 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.**

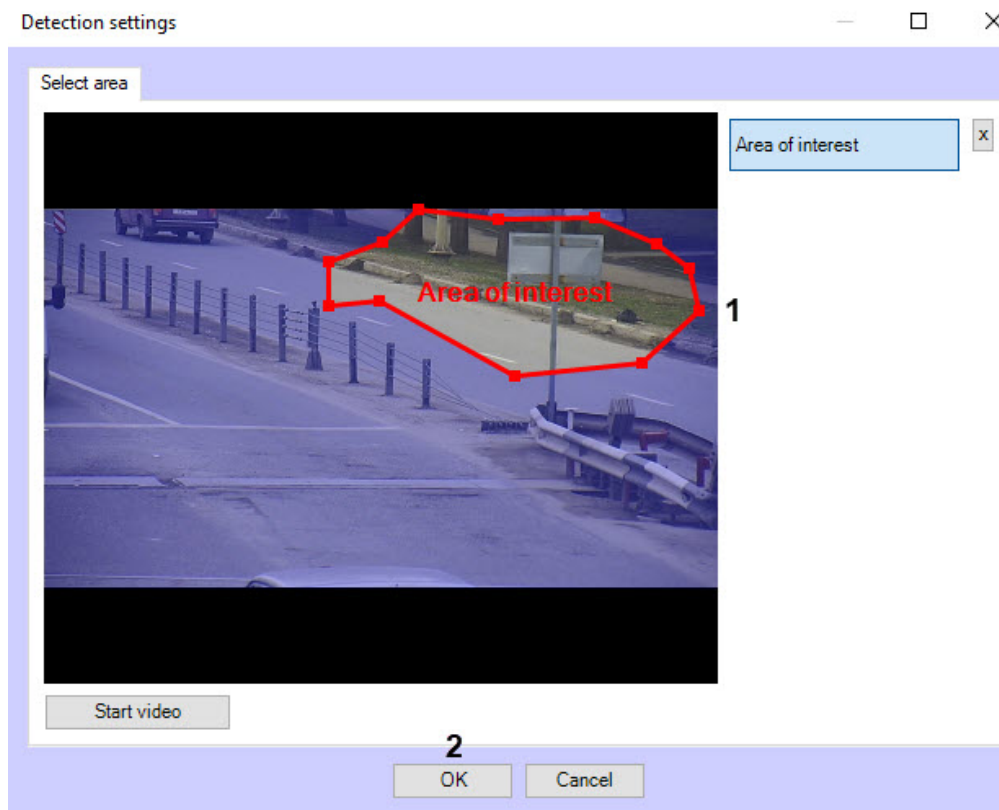
Multiplication of **Time in seconds between processed frames * Number of frames for analysis and output** parameter values characterizes time period (in seconds) at the end of which after fire/smoke alarm will be triggered.

9. In the **Device** drop-down list (8), select the device on which the neural network will operate.
10. Click the **Setup** button. The **Detection settings** windows will open.
11. Specify the 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 analyzed by clicking the left mouse button. 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.



- d. Click **OK** (2).

Note.

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

Attention!

Setting the **Area of interest** is a mandatory requirement for the detection module operation.

- Click the **Apply** button (10).

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

4.14 Fluid level detection

4.14.1 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:

- Determination of the fluid level in carriages passing in the field of view of the thermal camera.
- 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 recoded into the *Auto-Intellect* database.
- Putting titles indicating the fluid level onto the video image in the Video Surveillance Monitor.
- Putting the vertical line indicating the fluid level onto the video image in the Video Surveillance Monitor (using the script).
- 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.

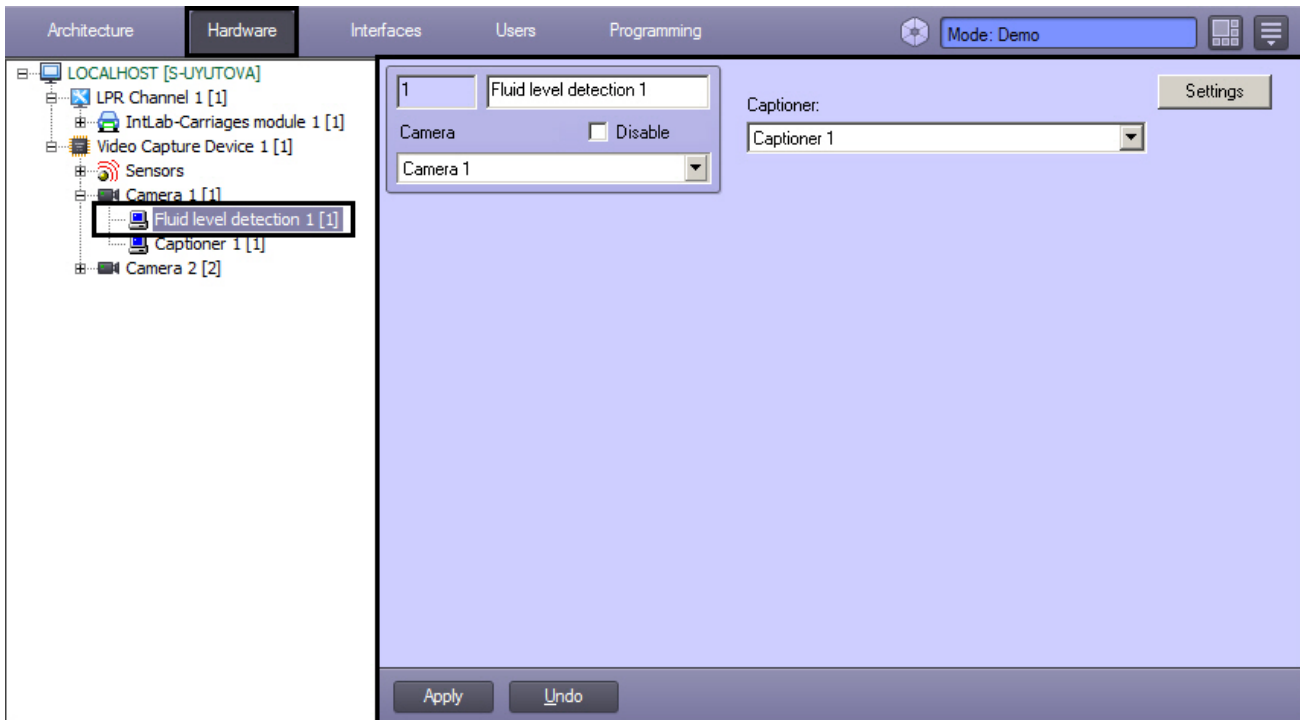
4.14.2 Camera requirements for the Fluid level detection module

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

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

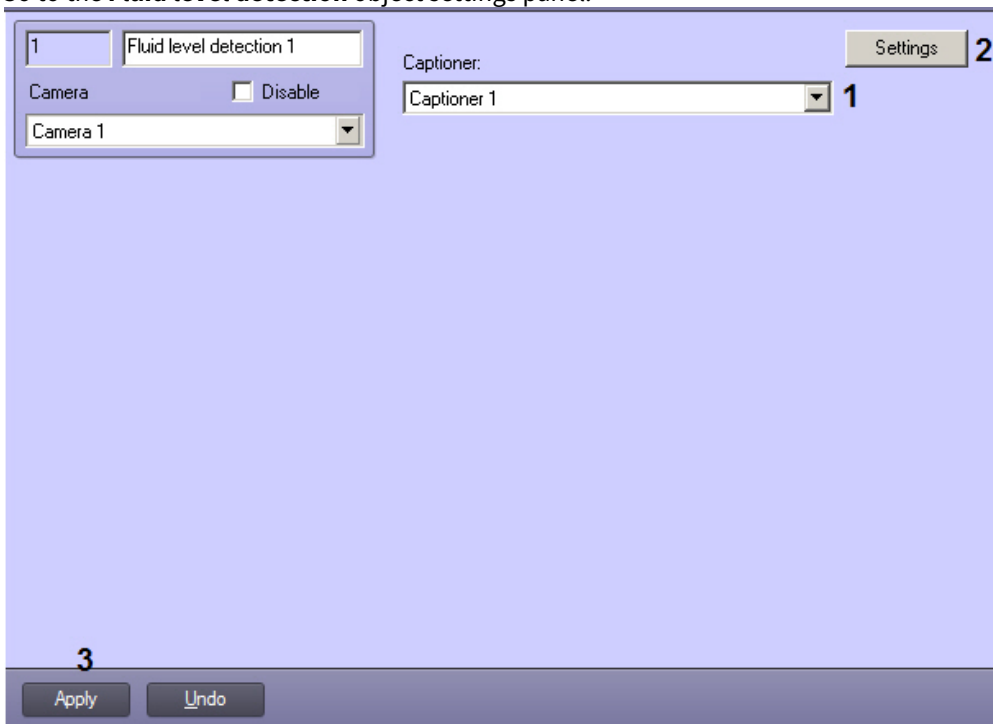
4.14.3 Configuring the Fluid level detection module

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.

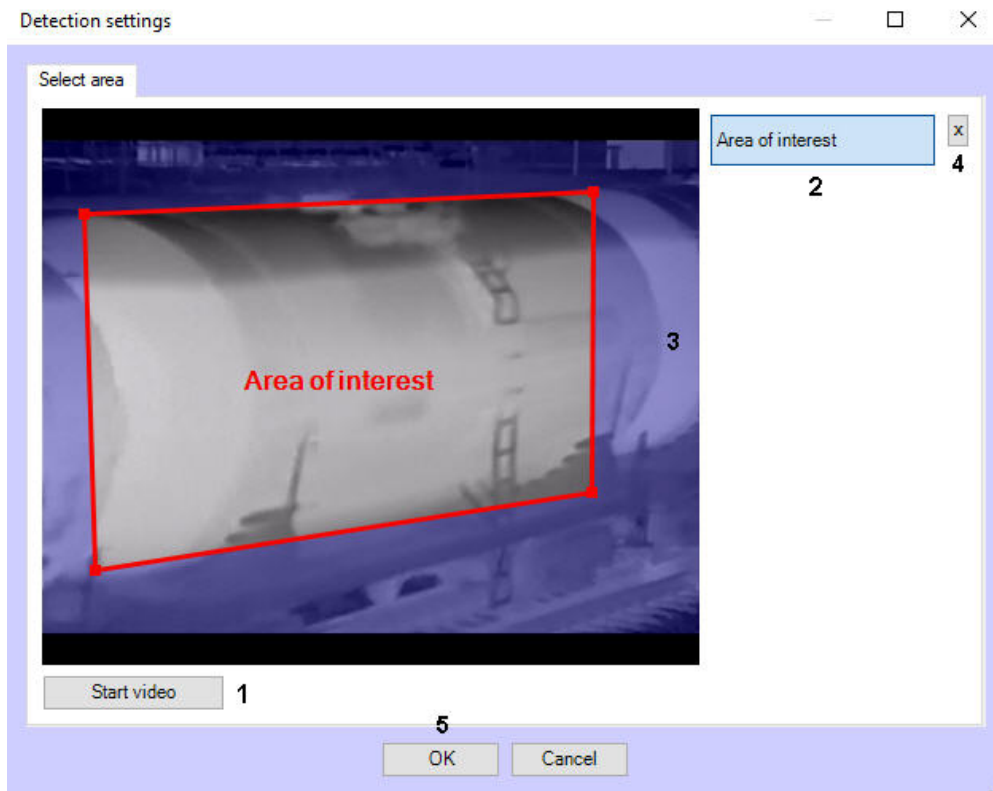


2. In the **Captioner** drop-down list select the **Captioner** object created under the same **Camera** object as the **Fluid level detection** object (1). This captioner will be used to overlay captions of fluid level onto the camera video image in the Video Surveillance Monitor.

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](#)).

4.15 Neurotracker

4.15.1 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).

4.15.2 Video stream and scene requirements for the Neurotracker module

The requirements for the video stream and the scene when working with the **Neurotracker** module are given in the table:

Video stream from camera	<ul style="list-style-type: none"> The resolution is at least 640x360 pixels. It is also not recommended to use a resolution higher than 1920x1080, since higher resolution does not increase the detection quality, but significantly increases the consumption of resources. The optimal resolution for solving typical tasks is 1280x720 (see Examples of configuring neural tracker for solving typical tasks). The frame rate per second in the video stream from the camera is at least 8 for solving typical tasks. Both colorless (gray) and color images.
Lighting	<ul style="list-style-type: none"> Lighting in the scene is at least 50 lux per square meter. In conditions of insufficient or excessive lighting (night or light-striking), stable operation of the video analytics is not guaranteed. There are no abrupt changes in lighting.
Scene and camera angle	<ul style="list-style-type: none"> Moving objects are visually separable from each other. The background is mostly static and does not change abruptly. Moving objects are minimally obscured by static objects in the scene (columns, trees, etc.). The analyzed scene does not have reflective surfaces and sharp shadows from moving objects. If present, they should be masked. Camera shake does not result in image offsets greater than 1% of the frame size.
Objects image	Objects image requirements for neural tracker

Attention!

Correct operation of the neural tracker is not guaranteed when using a fish-eye lens.

Software and hardware requirements


4.15.3 Objects image requirements for neural tracker

To ensure the correct operation of detection tools based on the neural tracker, the following image requirements should be met:


- The object to be detected is clearly distinguishable by the human eye.
- The width or height of the objects does not exceed 75% of the frame size.
- The image is not noisy and not distorted by compression algorithm artifacts.
- The duration of the object's visibility is at least 6 frames.
- The object moves in the certain direction between two adjacent frames at a distance which does not exceed the object's size. This condition is necessary for the correct calculation of the trajectory of the object (track).
- The minimum value of pixel density per meter is observed:

Image resolution	Object type	Minimum pixel density per meter (ratio of the object width in pixels to the object width in meters)	Minimum object size in pixels, Width x Height	Ratio of the object width to the frame width as a percentage
1920x1080	Human	55	~25x105	~3%
1280x720	Human	35	~17x70	~3%
640x360	Human	17	~10x42	~3%
1920x1080	Light vehicle (2 axles)	55	~354x300	~20%
1280x720	Light vehicle (2 axles)	35	~240x205	~20%
640x360	Light vehicle (2 axles)	17	~132x112	~2%

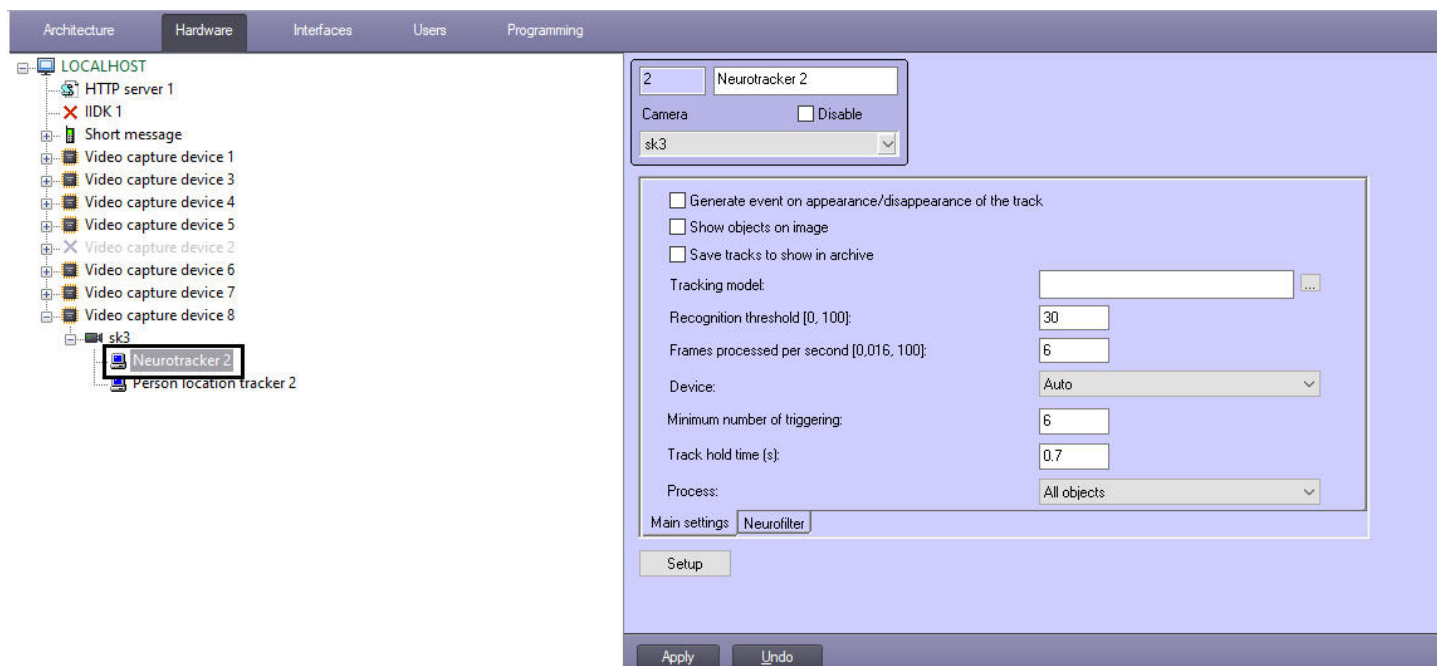
Image resolution	Object type	Minimum pixel density per meter (ratio of the object width in pixels to the object width in meters)	Minimum object size in pixels, Width x Height	Ratio of the object width to the frame width as a percentage
				0 %

 Video stream and scene requirements for the Neurotracker module

4.15.4 Configuring the Neurotracker module

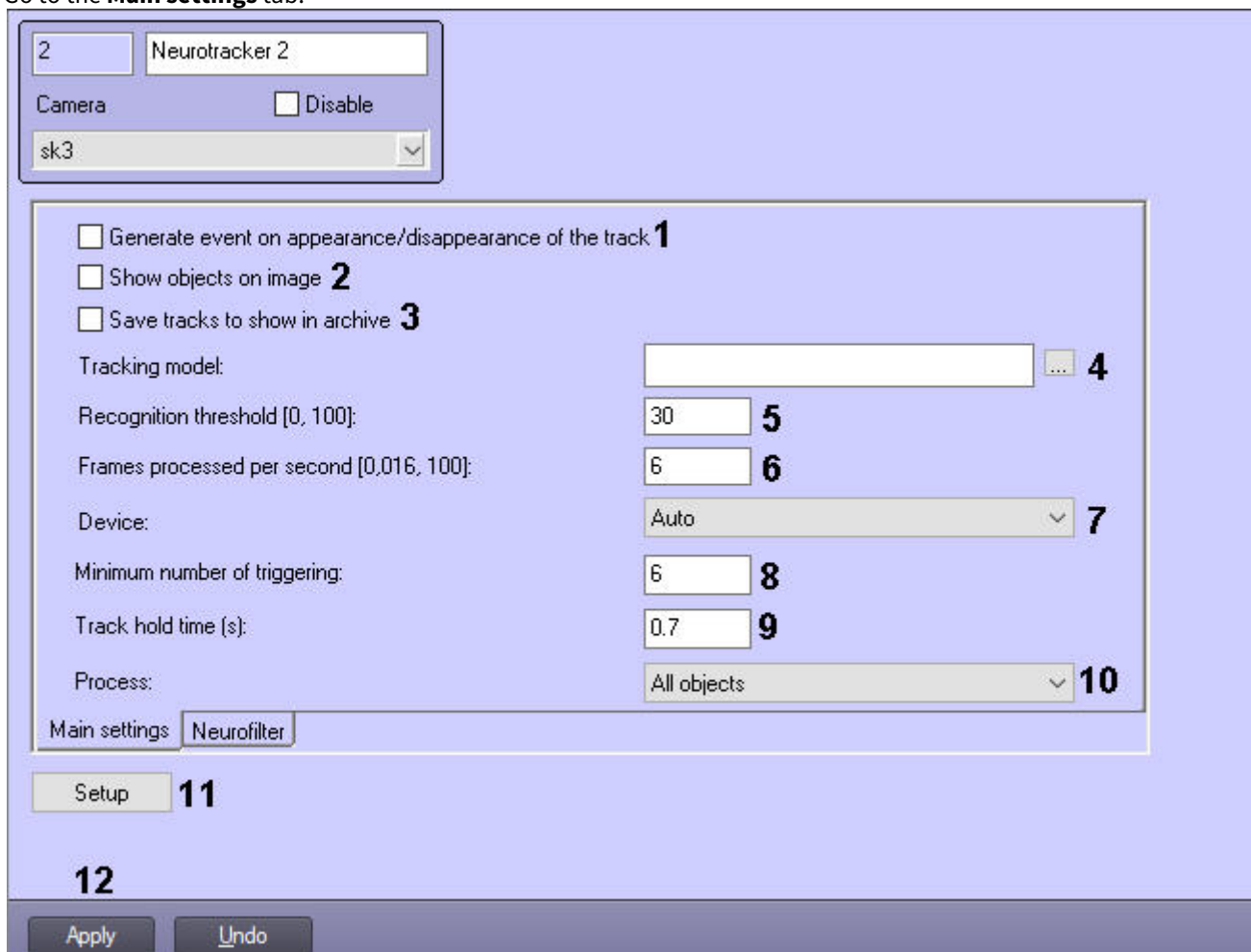
 **Attention!**
 The Neurotracker software module works only in *Intellect* 4.11.0 and higher version.

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 **Main settings** tab.



2. Set the **Generate event on appearance/disappearance of the track** 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 when viewing live video.
4. Set the **Save tracks to show in archive** checkbox (3), if it is necessary to highlight the detected object with a frame when viewing the archive.

Note

This parameter does not affect VMDA search and is used just for the visualization. For this parameter, the *titles* database is used.

5. Select the required neural network file with the tracking model (4).
6. In the **Recognition threshold [0, 100]** field (5), specify the objects detection threshold within the range from 0 to 100.

Note

The objects detection threshold is determined experimentally. The lower the detection threshold, the more false triggerings there might be. The higher the detection threshold, the less false triggerings there might be, however, some useful tracks might be skipped. See [Examples of configuring neural tracker for solving typical tasks](#).

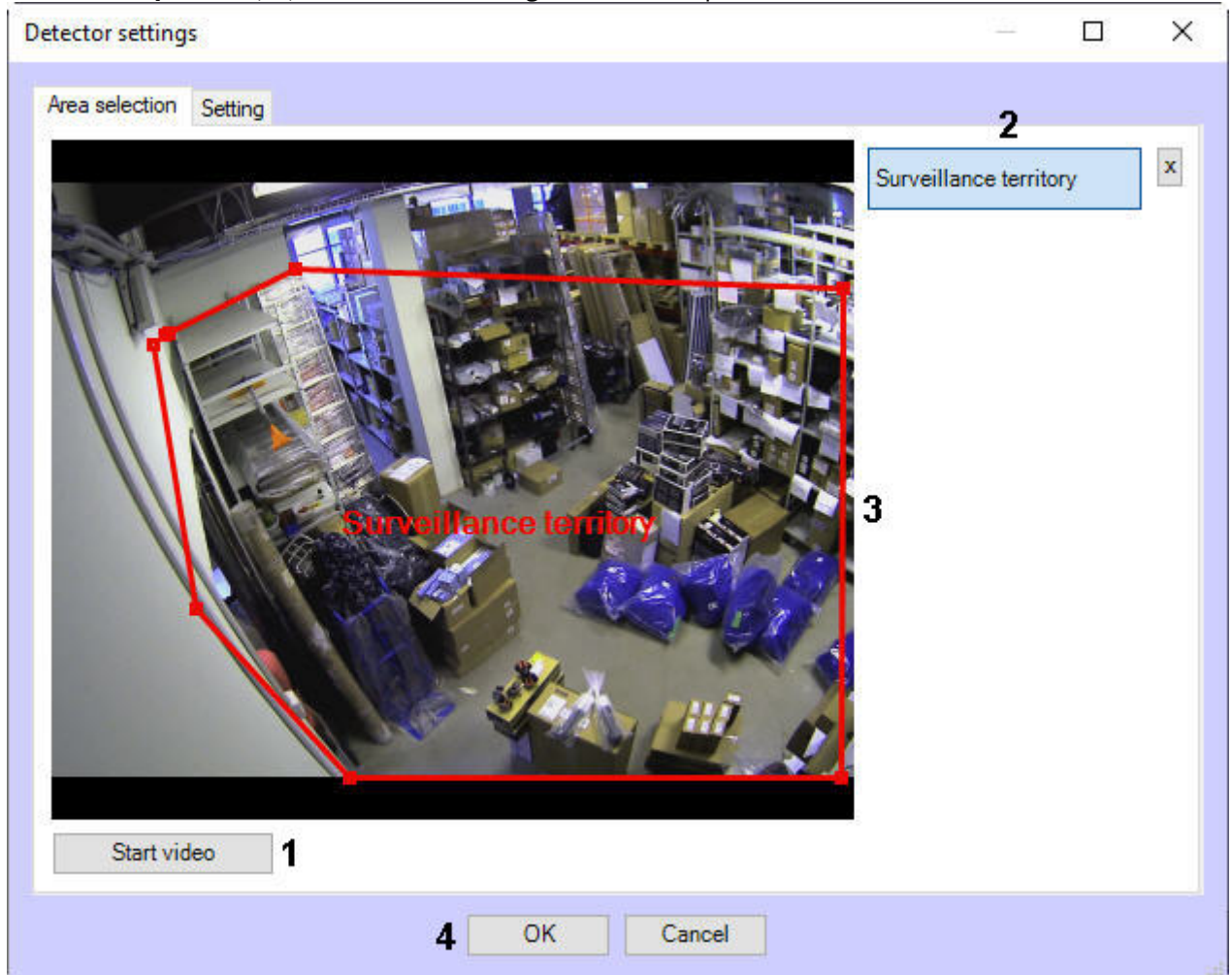
7. In the **Frames processed per second [0,016, 100]** field (6), set the number of frames per second that will be processed by the neural network. All other frames will be interpolated. The higher the specified value, the more accurate the tracking, but the higher the CPU load.
8. In the **Device** drop-down list (7) select the device where the neural network will operate.
9. In the **Minimum number of triggering** field (8), specify the minimum number of neurotracker triggers required to display the object track. The higher the value of this parameter, the longer it will take from the object detection moment to the display of its track. At the same time, a low value of this parameter can lead to false positives.
10. In the **Track hold time (s)** field (9), specify the time in seconds after which the object track is considered lost. This parameter can be useful in situations where one object in the frame temporarily overlaps another. For example, when a larger car completely overlaps a smaller one.

Note

If the object track is close to the frame boundary, then about a half of the time specified in the setting should elapse from the moment the object disappears from the frame until its track is deleted.

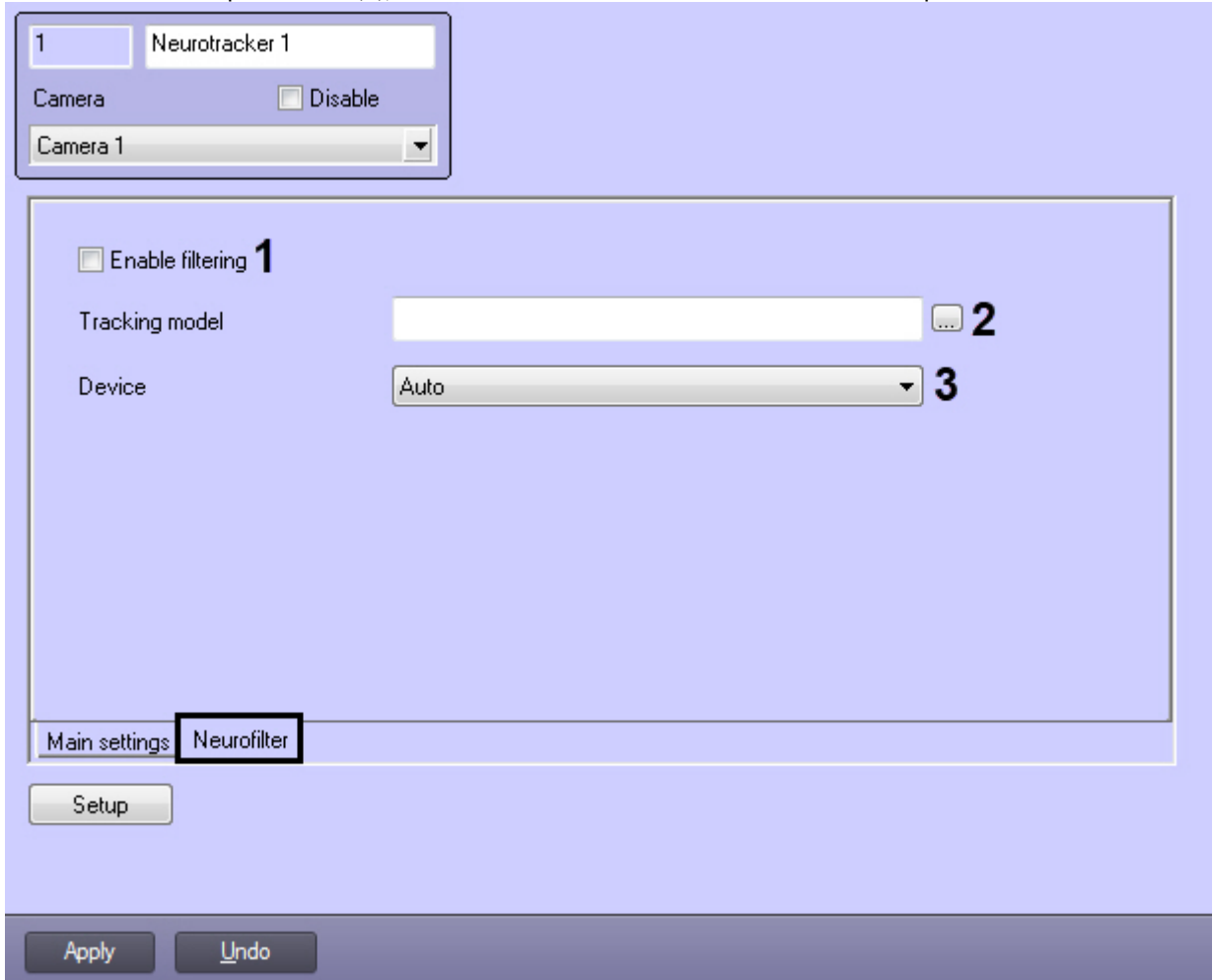
11. From the **Process** drop-down list (10) select which objects should be processed by the neural network:
 - **All objects** — moving and stationary objects.
 - **Only moving objects** — an object is considered to be moving if during the entire lifetime of its track it has shifted by more than 10% of its width or height. Using this parameter may reduce the number of false positives.
 - **Only stationary objects** — an object is considered stationary if during the entire existence of its track it has not shifted by more than 10% of its width or height.
12. Set the monitoring area on the video image:

- a. Click the **Setup** button (11). The **Detector settings** window will open.



- b. Click the **Start video** button (1) to capture a video frame.
- c. Click the **Surveillance territory** button (2).
- d. On the captured video frame, sequentially set the nodal points of the area in which the objects will be detected by left-clicking the mouse button (3).
- e. Click **OK** (4).
13. You can use the neural filter to sort out video recordings featuring selected objects and their trajectories. For example, the neural tracker detects all freight trucks, and the neural filter leaves only those tracks that correspond to trucks with cargo door open. To set up a neural filter, do the following:
- Go to the **Neurofilter** tab.
 - Set the **Enable filtering** checkbox (1).
 - Select the required neural network file for the neural filter (2).

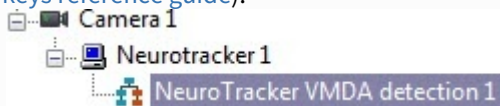
d. From the **Device** drop-down list (3), select the device on which the neural filter will operate.



14. Click the **Apply** button (12).

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](#)). Also, if you select the **Staying in the area for more than 10 sec** detector type, the time the object stays in the zone, after which the NeuroTracker VMDA detection tools are triggered, is configured using the LongInZoneTimeout2 registry key, not LongInZoneTimeout (see [Registry keys reference guide](#)).



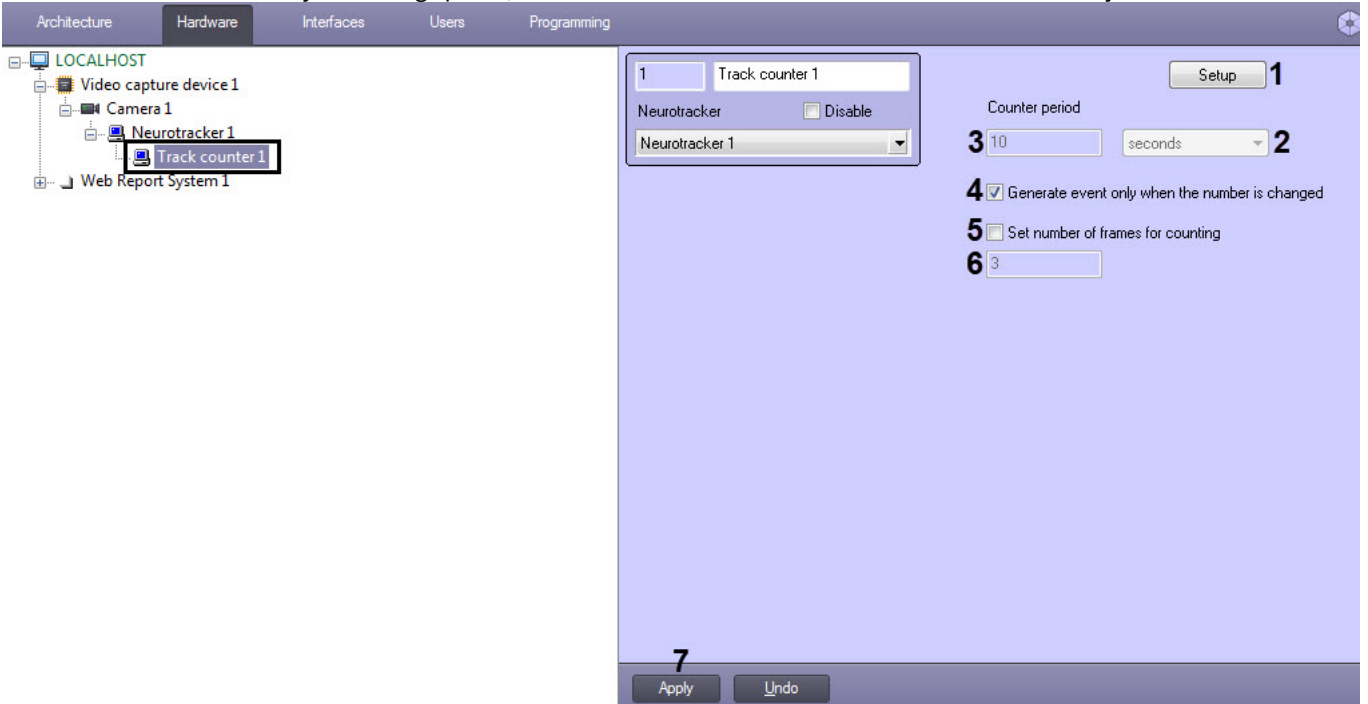
The **Neurotracker** software module configuration is complete.

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](#)).

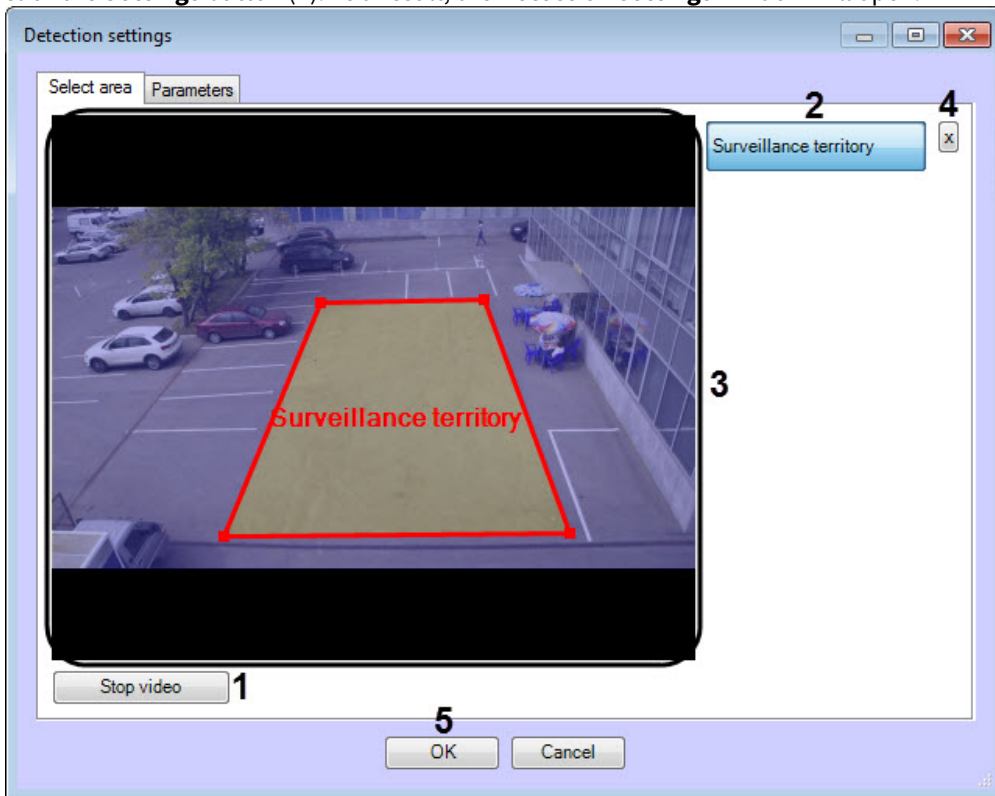
4.15.4.1 Configuring the neurotracker track counter

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. If the **Generate event only when the number is changed** check box is not set, then from the **Counter period** drop-down list (2) select the counter period time unit: **seconds, minutes, hours, days**. Then in the field (3), specify the time in which the event containing the total number of objects recorded for a given time period will be generated.
- 4. Set the **Generate event only when the number is changed** checkbox (4) if it is necessary to generate events only when the number of recorded objects changes.

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

- 5. If objects are lost during the track counter operation, then for more accurate counting, set the **Set number of frames for counting** checkbox (5) and in the field (6) enter the number of frames that will be analyzed to calculate the average objects number.

Note

- This feature works only if the **Generate event only when the number is changed** checkbox is set.
- The value of the number of frames for counting is selected experimentally depending on the neural network used. Recommended values are: 3, 5, 7, 9.

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

The neurotracker track counter is configured.

Note
 To display the events that were generated by the neurotracker track counter in the Event Viewer, it is necessary to configure a filter for the track counter. For details, see [Configuring event filters for displaying in the Event viewer](#).

4.15.4.2 Examples of configuring neural tracker for solving typical tasks

Recommended neurotracker settings for detecting moving people or vehicles are shown in the table:

Settings	Task: detection of moving people	Task: detection of moving vehicles
Main		
The number of frames processed per second	6	12
Minimum number of triggerings	6	6
Neural filter	no	no
Recognition threshold	30	30
Tracking model	Path to the *.ann neural network file. Or you can select a specific Object type .	Path to the *.ann neural network file. Or you can select a specific Object type .
Objects to process	Only moving objects	Only moving objects

Note
 By default, the neural tracker is configured for detection of moving people.

4.16 Person location tracker

4.16.1 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.

4.16.2 Video stream and scene requirements for Person location tracker

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

Video stream from camera	<ul style="list-style-type: none"> • The resolution is at least 640x360 pixels. It is also not recommended to use a resolution higher than 1920x1080, since higher resolution does not increase the detection quality, but significantly increases the consumption of resources. The optimal resolution for solving typical tasks is 1280x720. • The frame rate per second in the video stream from the camera is at least 8. • Color image only.
Lighting	<ul style="list-style-type: none"> • Lighting in the scene is at least 50 lux per square meter. In conditions of insufficient or excessive lighting (night or light-striking), stable operation of the video analytics is not guaranteed. • There are no abrupt changes in lighting.
Scene and camera angle	<ul style="list-style-type: none"> • Moving objects are visually separable from each other. • The background is mostly static and does not change abruptly. • Moving objects are minimally obscured by static objects in the scene (columns, trees, etc.). • The analyzed scene does not have reflective surfaces and sharp shadows from moving objects. If present, they should be masked. • Camera shake does not result in image offsets greater than 1% of the frame size.
Objects image	Objects image requirements for Person location tracker

Attention!

Correct operation of the pose detection tools is not guaranteed when using a fish-eye lens.

[Software and hardware requirements](#)

4.16.3 Objects image requirements for Person location tracker

To ensure the correct recognition of a person's posture, the following image requirements should be met:

1. The object to be detected is clearly distinguishable by the human eye.
2. The detected object is completely within the frame.
3. The image is not noisy and not distorted by compression algorithm artifacts.

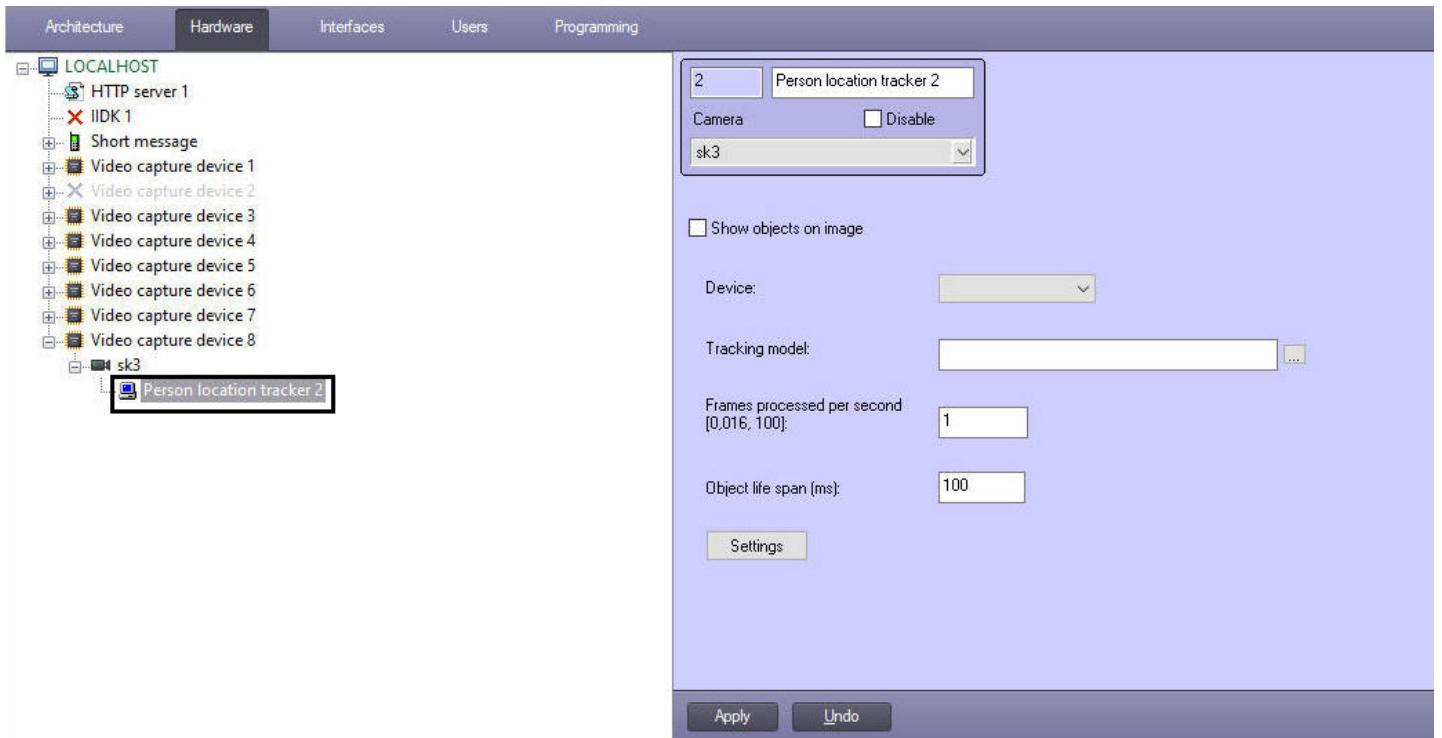
- 4. The duration of the object's visibility is at least 2 frames.
- 5. The minimum value of pixel density per meter is observed:

Image resolution	Object type	Minimum pixel density per meter (ratio of the object width in pixels to the object width in meters)	Minimum object size in pixels, Width x Height
For pose detection tools			
1920x1080	Human	170	~102x309
1280x720	Human	128	~77x233
640x360	Human	80	~48x145

Video stream and scene requirements for Person location tracker

4.16.4 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 when viewing live video.
3. From the **Device** drop-down list (2) select the device where the neural network will operate.
4. Select the required neural network file with the tracking model (3).
5. In the **Frames processed per second [0,016, 100]** field (4), set the number of frames per second that will be processed by the detection tool.

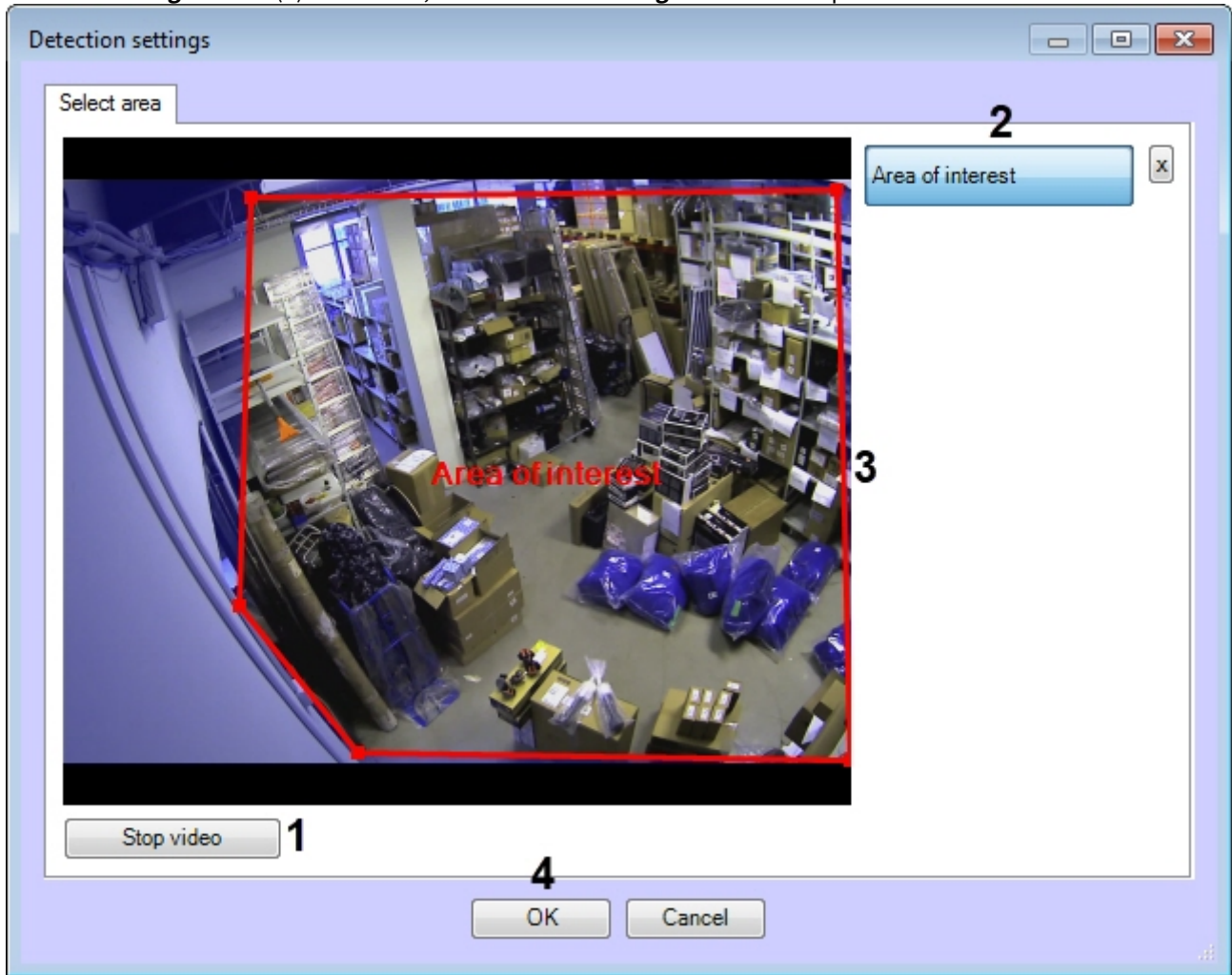
⚠ Attention!

For scene with static people, the fps value should be at least 2. For scene with moving people — at least 4. The larger this value, the higher the accuracy of pose detection, but the greater the CPU load. With fps=1, the accuracy will be no less than 70%. This parameter varies depending on the object speed of movement. To solve typical tasks, fps value from 3 to 20 is sufficient. Examples:

- pose detection for moderately moving objects (without sudden movements) — fps=3;
- pose detection for moving objects — fps=12.


6. In the **Object life span (ms)** field (5), enter the time in milliseconds in which the found object will be considered as disappeared from the frame.

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

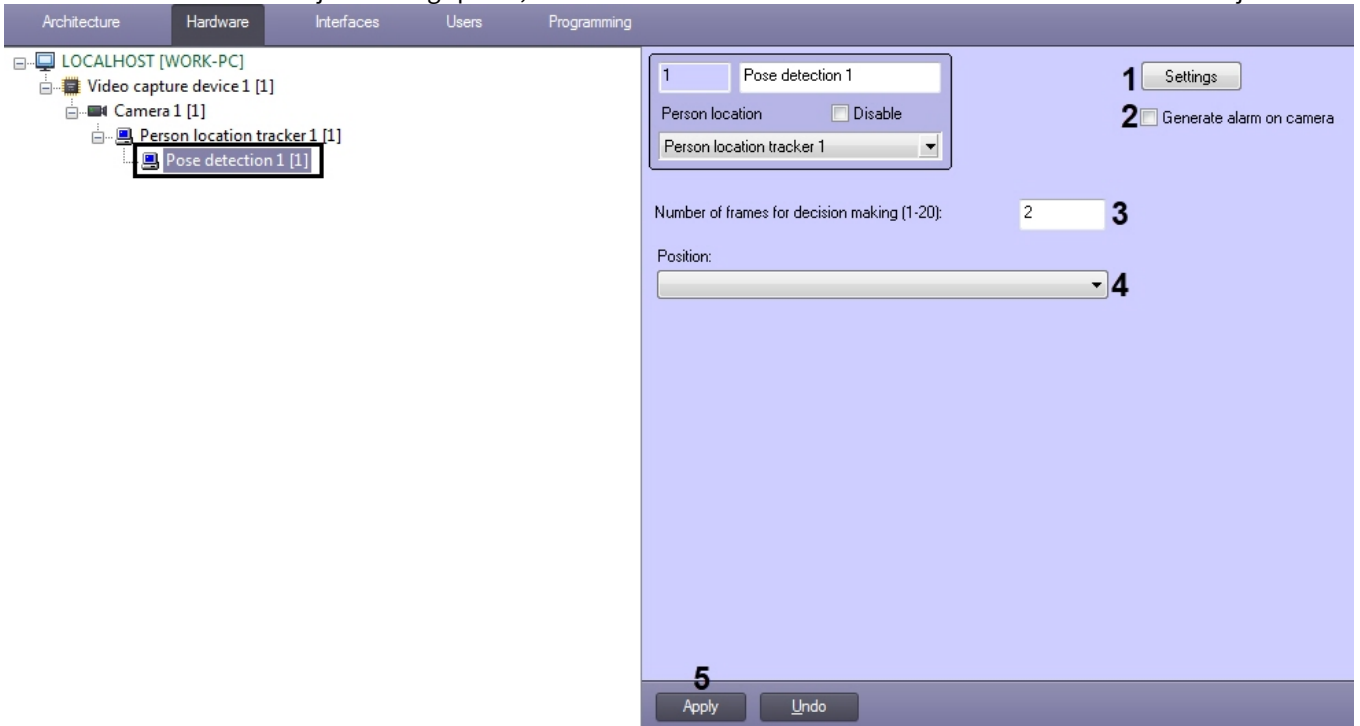
To delete the selected area, click the  button.

- c. Click **OK** button (4).
8. Click the **Apply** button (7).
 9. Perform pose detection configuration (see [Configuring the pose detection](#)).

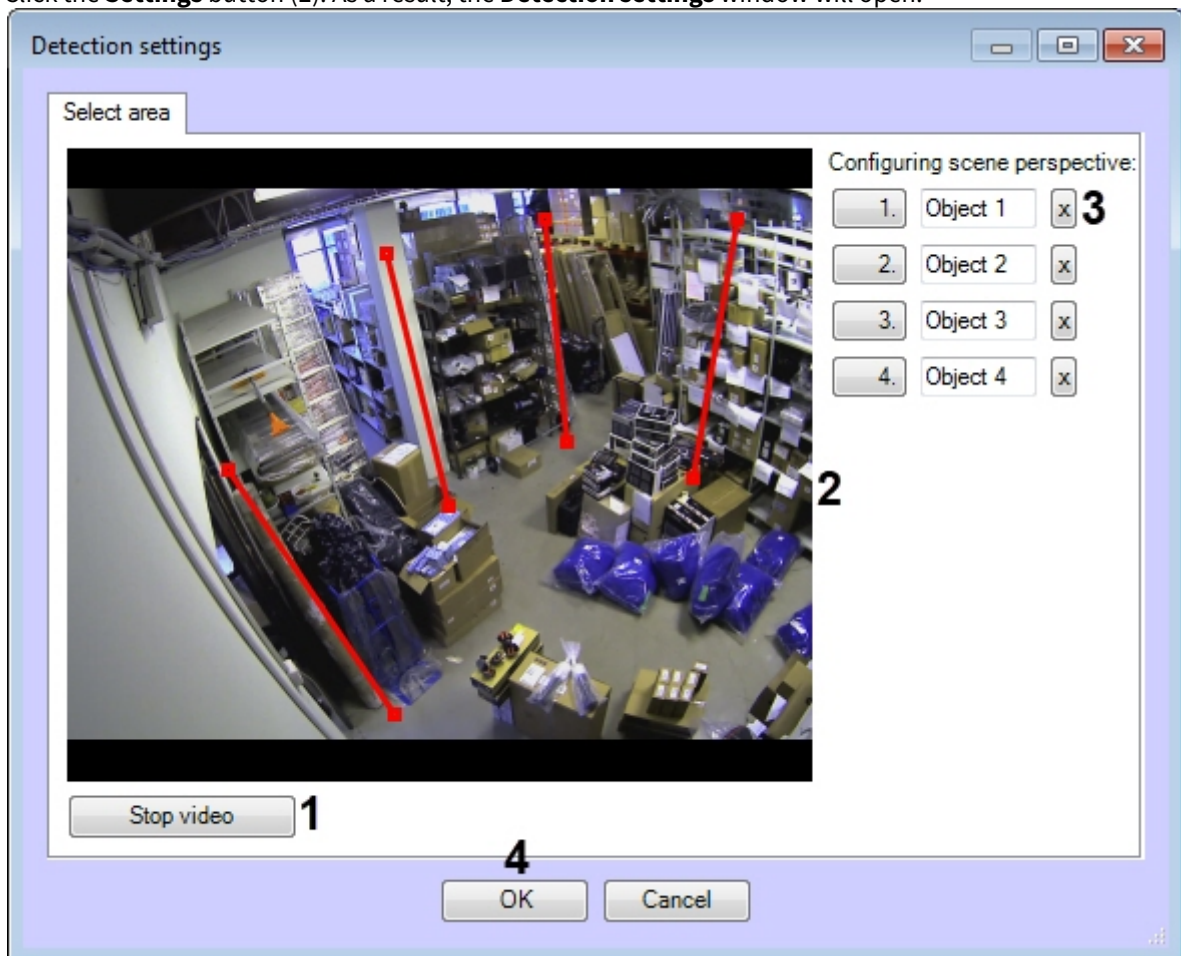
4.16.4.1 Configuring the pose detection

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. Specify the perspective on the video image on which the human location will be detected:
 - a. Click the **Settings** button (1). As a result, the **Detection settings** window will open.



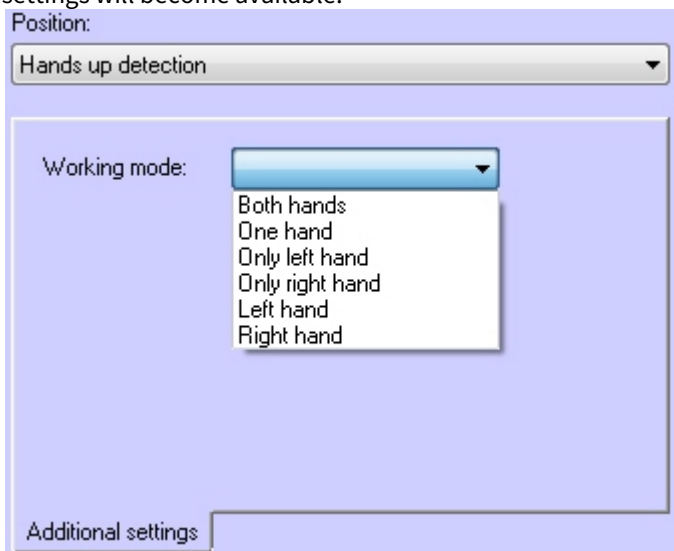
- b. Click the **Stop video** button (1) to capture the video image.

- c. Specify the size of the same human at least in three different parts of the frame (2) with the calibration segments. To create a calibration segment, click and hold the left mouse button on the video image so that you can specify the necessary length size. You can change the size of the already created segments using nodes. The calibration segments should be located at different angles to each other depending on the scene and the distortion of the video image.

Note

To delete the created segment, click the  button (3).

- d. Click **OK** (4).
- 3. Set the **Trigger an alarm on camera** checkbox (2) if it is also necessary to trigger an alarm on the camera by the specified pose detection.
- 4. In the **Number of frames for decision making (1-20)** field (3), enter the required amount of frames from 1 to 20 on which the human should be found in the specified pose to trigger the corresponding event.
- 5. From the **Type** drop-down list (4), 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 hands raised is found in the frame. The hand is considered to be raised if the forearm is parallel to the backbone. If this type is selected, the following additional settings will become available:



- **Working mode** — sets the hands position at which the alarm event will be generated.
- **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. If this type is selected, the following additional settings will become available:

- **Calibration column length [0, 3]** — sets the length of the calibration segments, which were set when adjusting the perspective of the video image, from 0 to 3 meters.
- **Distance sensitivity [0, 20]** — sets the minimum distance between people from 0 to 20 meters. If the distance between people in the frame is equal to or less than the specified one, then the detection will be triggered.
- **Number of people to alert [2, 20]** — sets the number of people who should break the distance for the detection to be triggered. Possible values: from 2 to 20.

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

The pose detection is now configured.

4.17 VideoIntellect embedded detector

4.17.1 Functionality of the VideoIntellect embedded detector module

The VideoIntellect embedded detector module includes the following detection tools:

- **Abandoned objects detection** - designed for use as part of video surveillance systems at crowded objects (metro, train stations, airports, museums, shopping malls, etc.) in order to automatically detect abandoned or disappeared objects, items, things, and other stationary scene changes that occurred in the surveillance area in the camera field of view and remain motionless for a given period of time.
- **Prohibited zone motion detection** - designed for use as part of video surveillance systems at crowded objects (metro, train stations, airports, museums, shopping malls, etc.) in order to automatically detect people movement in the camera field of view.
- **Camera state detection** - designed for automatic detection of atypical changes in the scene (flare, darkening, defocusing).
- **Prohibited direction motion detection** - designed for use as part of video surveillance systems at crowded objects (metro, train stations, airports, museums, shopping malls, etc.) in order to automatically detect people movement in a prohibited direction in the camera field of view.
- **Other** - other VideoIntellect detection tools that are not integrated at the moment (see the official [VideoIntellect website](#)).

4.17.2 Licensing of the VideoIntellect embedded detector module

The VideoIntellect embedded detector module is licensed using an executable *.exe file that includes a license for each individual detection tool. To obtain this license file, please contact VideoIntellect technical support ([official website](#)).

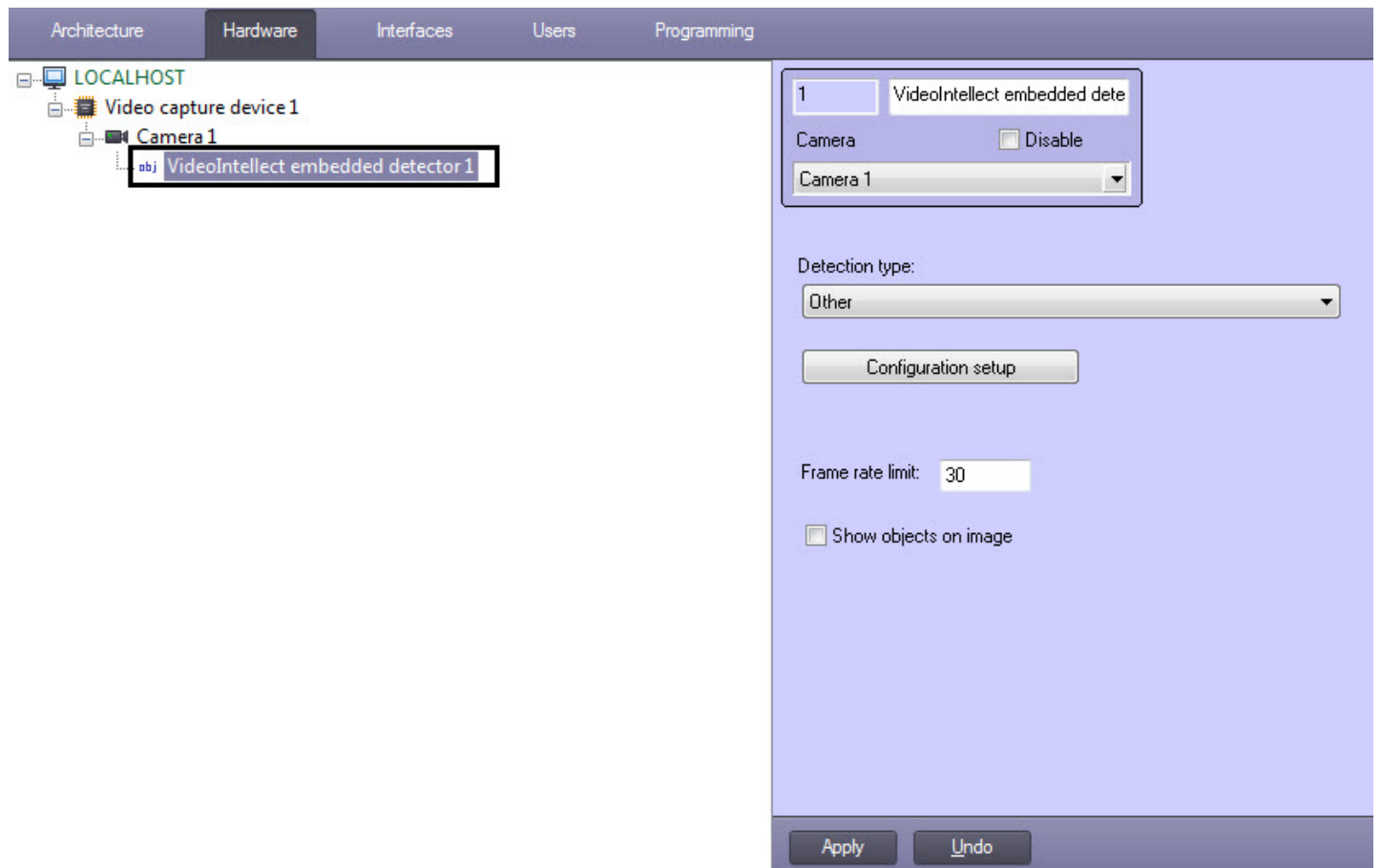
4.17.3 Camera requirements for the VideoIntellect embedded detector module

The requirements for cameras that will work with the **VideoIntellect embedded detector** module are shown in the following table.

Camera	<ul style="list-style-type: none"> Resolution: at least 640x480 pixels Correct operation is not guaranteed if a fish-eye lens is used Fps: not less than 12
Lighting	<ul style="list-style-type: none"> The objects should be visually separated from the background and from each other
Scene and camera angle	<ul style="list-style-type: none"> The camera should be static

4.17.4 Configuring the VideoIntellect embedded detector module

The **VideoIntellect embedded detector** module is configured on the settings panel of the **VideoIntellect embedded detector** object created on the basis of the **Camera** object on the **Hardware** tab of the **System settings** dialog window.



The **VideoIntellect embedded detector** module is configured as follows:

1. Go to the **Videointellect embedded detector** object settings panel.

2. From the **Detection type** drop-down list (1) select the required detection tool:
 - Abandoned objects detection.
 - Prohibited zone motion detection.
 - Camera state detection.
 - Prohibited direction motion detection.
 - Other.
3. Click the **Configuration setup** button (2) to configure the selected detection tool. As a result, the **Detection settings** window will open (for details, see [Configuring Videointellect detection tools](#)).
4. In the **Frame rate limit** field (3), enter the maximum number of frames per second that will be processed by the selected detection tool.
5. Set the **Show objects in image** checkbox (4) if it is necessary to highlight the detected object with a frame on the image in the **Monitor** interface object window.
6. Click the **Apply** button (5).

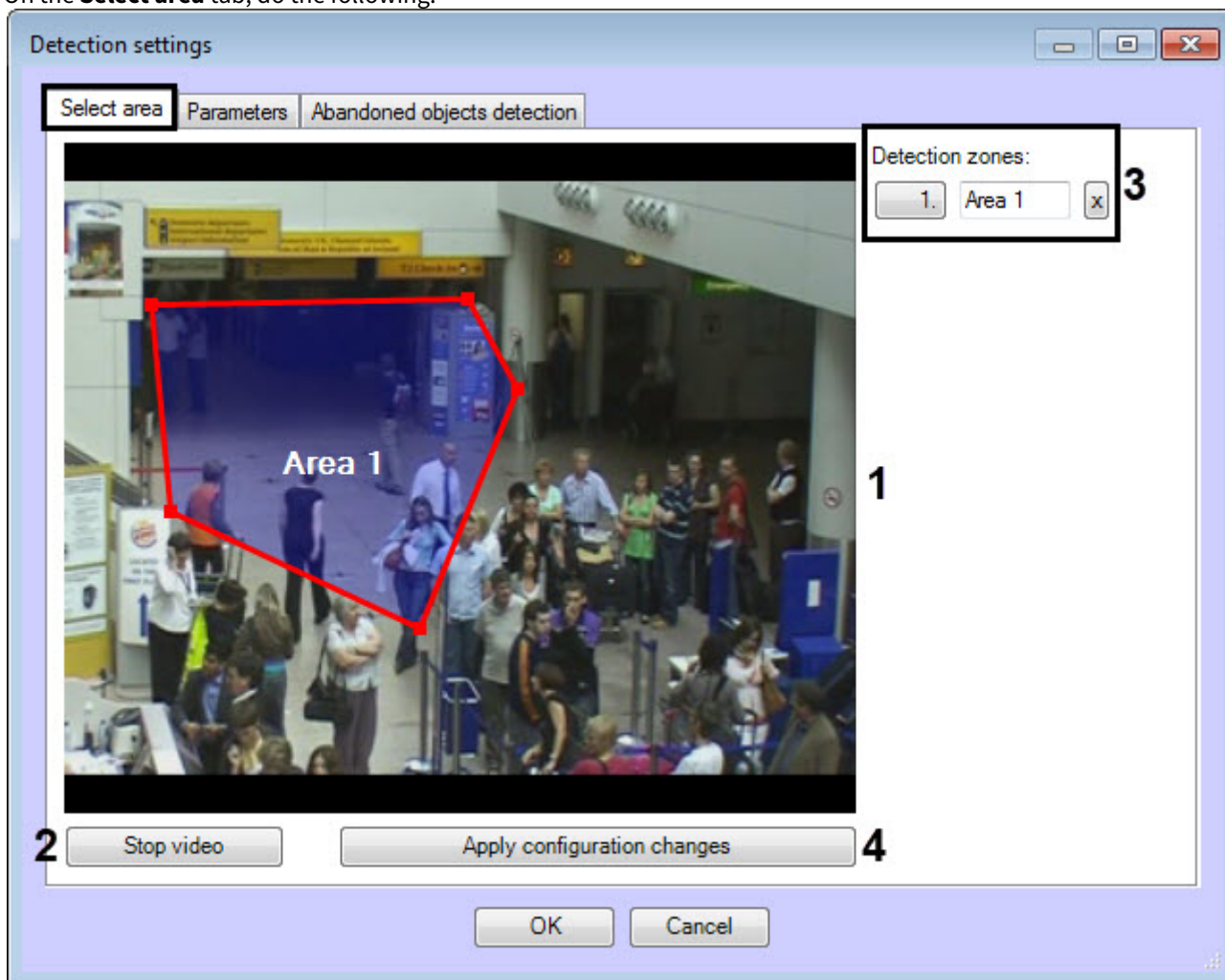
The **Videointellect embedded detector** module is now configured.

4.17.4.1 Configuring Videointellect detection tools

4.17.4.1.1 Videointellect abandoned objects detection

The Videointellect abandoned objects detection is configured as follows:

1. On the **Select area** tab, do the following:



- a. Set one or more zones in which the abandoned objects will be detected (1).
- b. To capture a frame of a video image, click the **Stop video** button (2).
- c. All set zones are displayed in the **Detection zones** area (3). If necessary, you can change the name of the zone or delete the zone.
- d. Click the **Apply configuration changes** button (4).

2. On the **Abandoned objects detection** tab, configure the detection tool parameters:

Detection settings

Select area Parameters **Abandoned objects detection**

Initial detection sensitivity (0-1): 0.00 **1**

Object visibility (0-1): 0.00 **2**

The passenger traffic density (0-1): 0.00 **3**

Minimum size of detected object (cm): 0 **4**

Maximum size of detected object (cm): 0 **5**

Detection time (s): 0 **6**

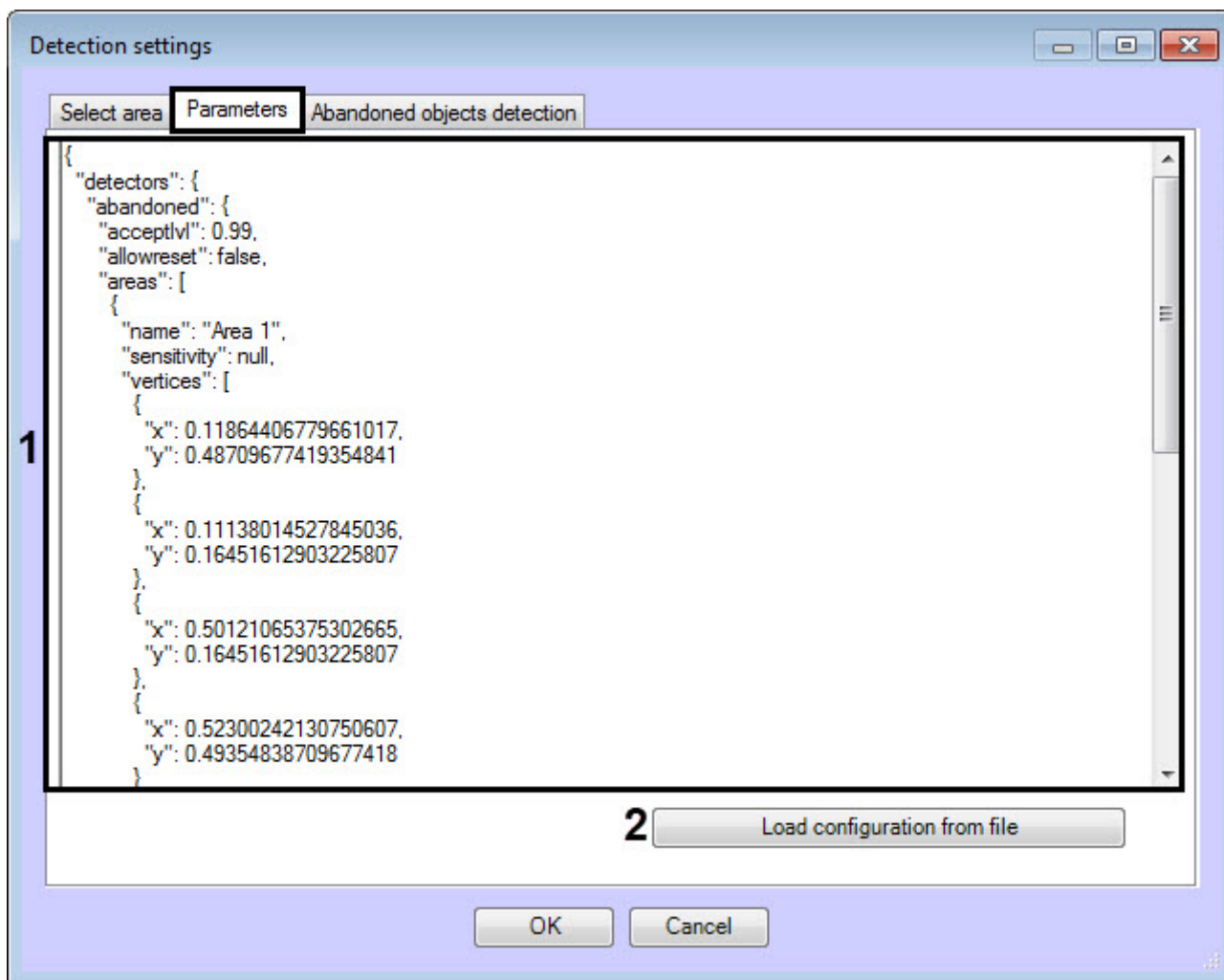
Shadow/highlight filter type: **7**

Without person: **8**

Filter objects: **9**

OK Cancel

- In the **Initial detection sensitivity (0-1)** field (**1**), enter the general sensitivity of the detector. Value range is from 0 to 1. Value 0 means that no object will be detected, value 1 means maximum sensitivity. The recommended value is 0.7.
 - In the **Object visibility (0-1)** field (**2**), enter the detection sensitivity of subtle (merging with the background) objects. This parameter indicates the ability of the detector to distinguish subtle (merging with the background) objects. Value range is from 0 to 1 in conventional units. Value 0 means the object will not be detected, value 1 means maximum sensitivity. The recommended value is 0.95. The recommended value range is from 0.82 to 0.97.
 - In **The passenger traffic density (0-1)** field (**3**), enter the object detection sensitivity in conditions of intensive object overlap by passing people.
 - In the **Minimum size of detected object (cm)** field (**4**) enter the minimum object size on one side in centimeters to be detected. Smaller objects will not trigger the detection. The size of the object is specified relative to the frame.
 - In the **Maximum size of detected object (cm)** field (**5**), enter the maximum object size on one side in centimeters to be detected. Larger objects will not trigger the detection. The size of the object is specified relative to the frame.
 - In the **Detection time (s)** field (**6**) enter the time in seconds after which the object is considered abandoned.
 - In the **Shadow/highlight filter type** drop-down list (**7**), enable additional filtering of shadows and highlights (disabled by default). Select one of the filter types: "z003", "z002", "z001", "zac", "zu", "zz" depending on the scene contrast, lighting variability, presence of shadows and highlights. For outdoor areas with shadows and visual noise (caused by grass and other plants, for example), and for objects with glares, the filter types "z003" and "zz" are recommended. For underbridge spaces with shadows and contrasting objects, the filter types are "z003" and "zac" are recommended. The filter is selected empirically.
 - Set the **Without person** checkbox (**8**) to ignore the objects left next to the person.
 - Set the **Filter objects** checkbox (**9**) to enable the object neural network filter.
3. On the **Parameters** tab, the area (**1**) displays the current detection configuration. This configuration can be copied to any text file. To upload a configuration from a file, click the **Load configuration from file** button (**2**).



Description of configuration parameters:

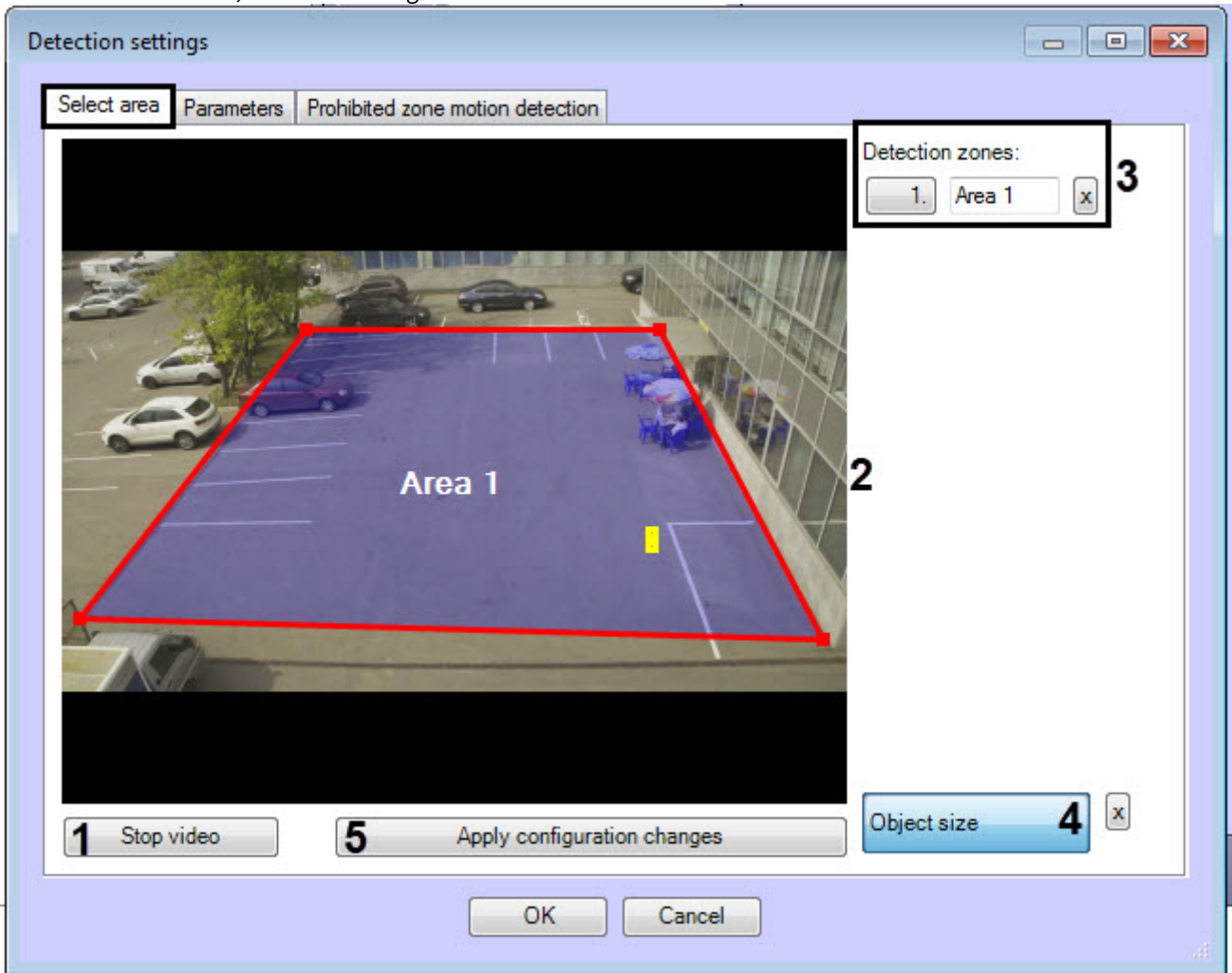
- a. sensitivity — Initial detection sensitivity.
 - b. rejectlvl — Object visibility.
 - c. acceptlvl — The passenger traffic density.
 - d. objsizemin — Minimum size of detected object.
 - e. objsizemax — Maximum size of detected object.
 - f. detectiontime — Detection time.
 - g. isFilterHumans — Without person.
 - h. isFilterObjects — Filter objects.
4. Click the **OK** button to complete the detection configuration.

The Videointellect abandoned objects detection is now configured.

4.17.4.1.2 Videointellect prohibited zone motion detection

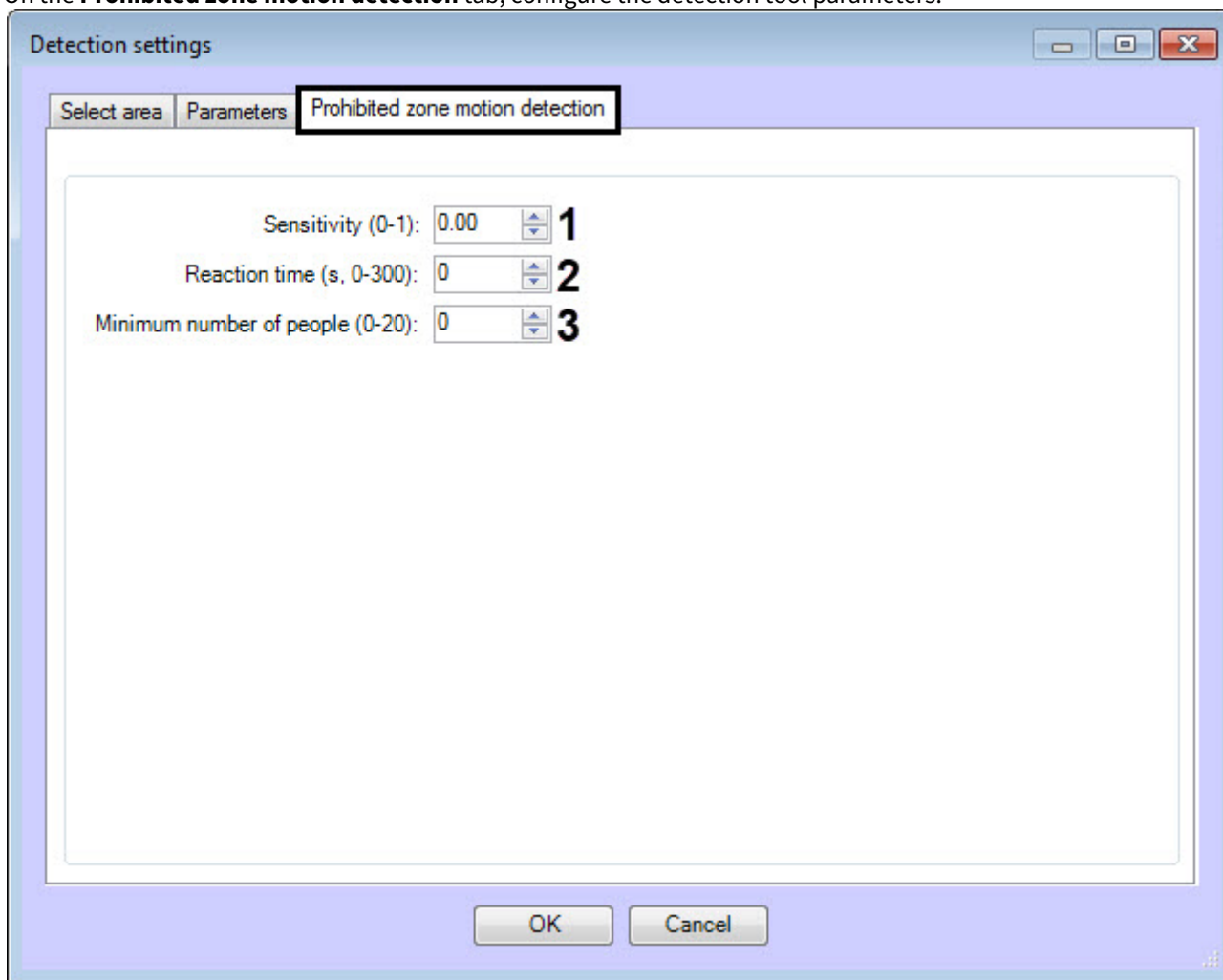
The Videointellect prohibited zone motion detection is configured as follows:

1. On the **Select area** tab, do the following:

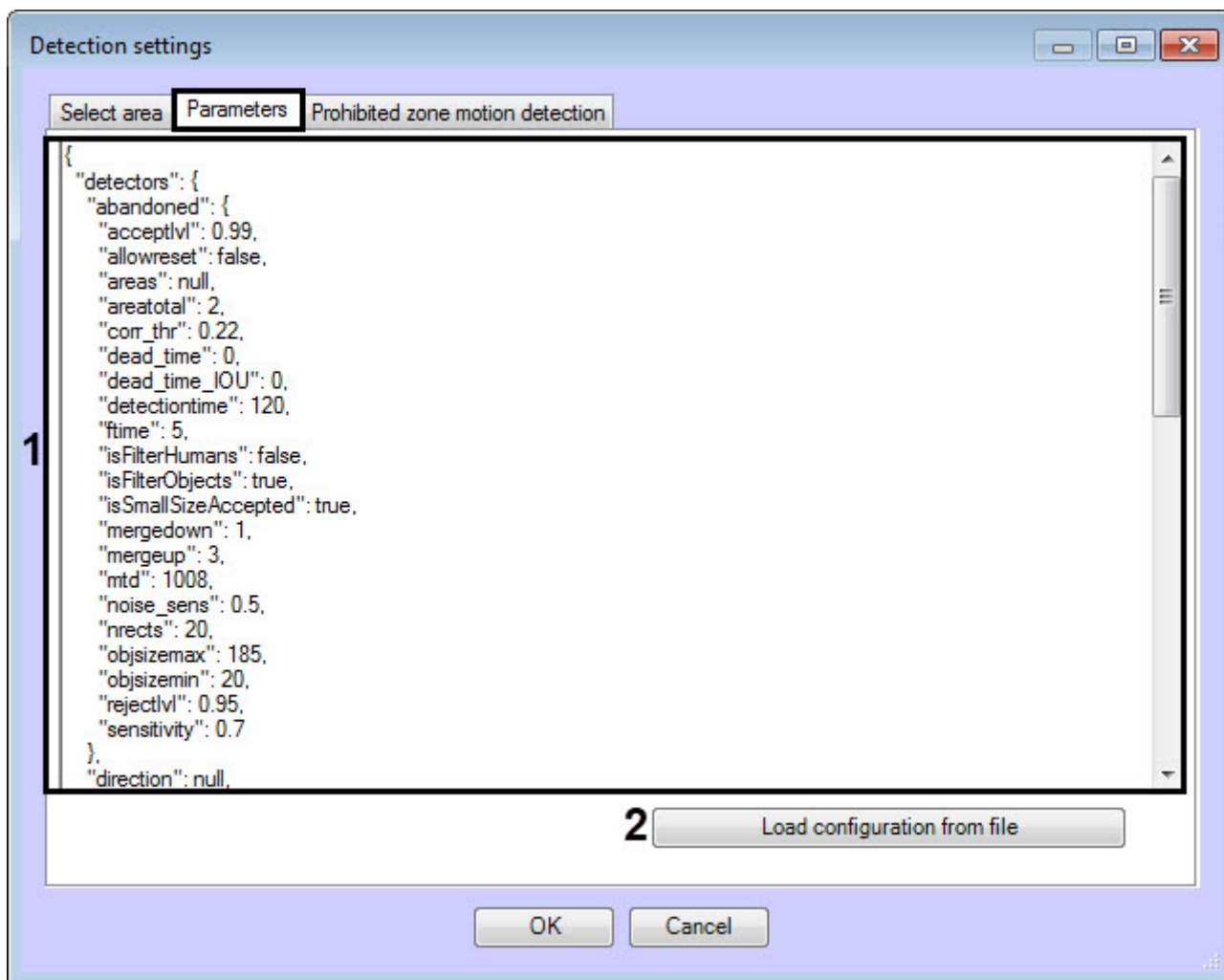


- a. To capture a frame of a video image, click the **Stop video** button (1).
- b. Set one or more zones in which the motion will be detected (2).
- c. All set zones are displayed in the **Detection zones** area (3). If necessary, you can change the name of the zone or delete the zone.
- d. Click the **Object size** button (4) and use the yellow line to set the approximate size of the human head.
- e. Click the **Apply configuration changes** button (5).

2. On the **Prohibited zone motion detection** tab, configure the detection tool parameters:



- a. In the **Sensitivity (0-1)** field (1), enter the detection tool sensitivity to detect moving objects in the analyzed area. The higher the sensitivity, the less visible a moving object can be detected.
 - b. In the **Reaction time (s, 0-300)** field (2), enter the time in seconds after which the detection tool will indicate the occurrence of the "Prohibited zone motion detection" alarm situation. The countdown starts from the moment the object enters the detection zone. If there is no movement in the prohibited zone after the reaction time has expired, the time counting stops, and the time counter is set to zero.
 - c. In the **Minimum number of people (0-20)** field (3), enter the minimum number of people at which the detection tool starts recording an event. If the zone occupancy with people continues to be higher than the specified threshold time (**Reaction time** parameter), then the detection tool indicates an alarm situation.
3. On the **Parameters** tab, the area (1) displays the current detection configuration. This configuration can be copied to any text file. To upload a configuration from a file, click the **Load configuration from file** button (2).



Description of some configuration parameters:

- a. sensitivity - Sensitivity.
 - b. sizeobject - The size of the human head.
 - c. reactiontime - Reaction time.
 - d. minNumberOfObjects - Minimum number of people.
4. Click the **OK** button to complete the detection configuration.

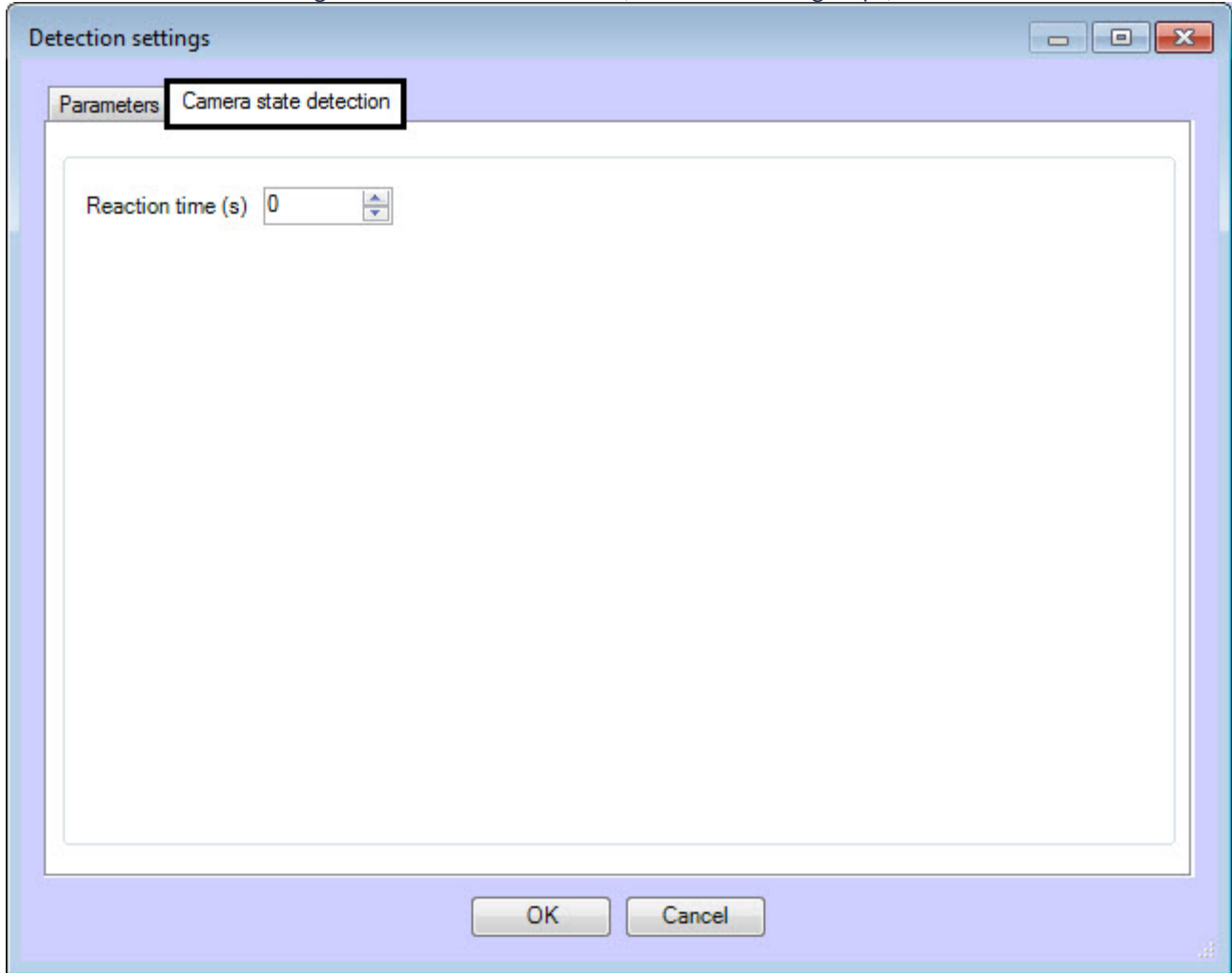
The Videointellect prohibited zone motion detection is now configured.

4.17.4.1.3 Videointellect camera state detection

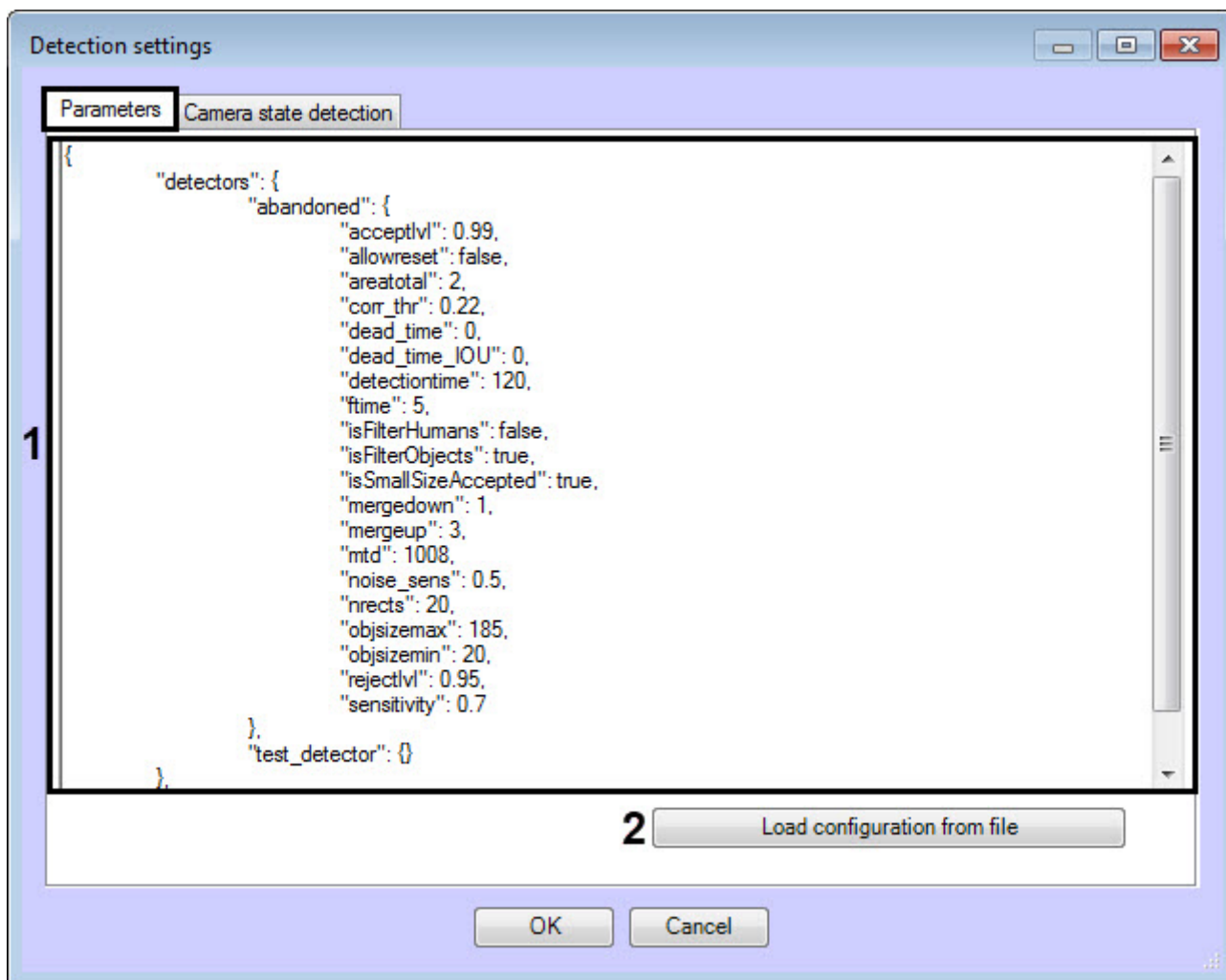
The Videointellect camera state detection is configured as follows:

1. On the **Camera state detection** tab, in the **Reaction time (s)** field, enter the time in seconds after which the detection tool indicates the fact of an atypical change in the scene. The countdown starts from the moment an alarm situation is

detected. If there are no changes in the state of the camera, the time counting stops, and the time counter is set to zero.



2. On the **Parameters** tab, the area (1) displays the current detection configuration. This configuration can be copied to any text file. To upload a configuration from a file, click the **Load configuration from file** button (2).



Description of configuration parameters:

a. reactiontime - Reaction time.

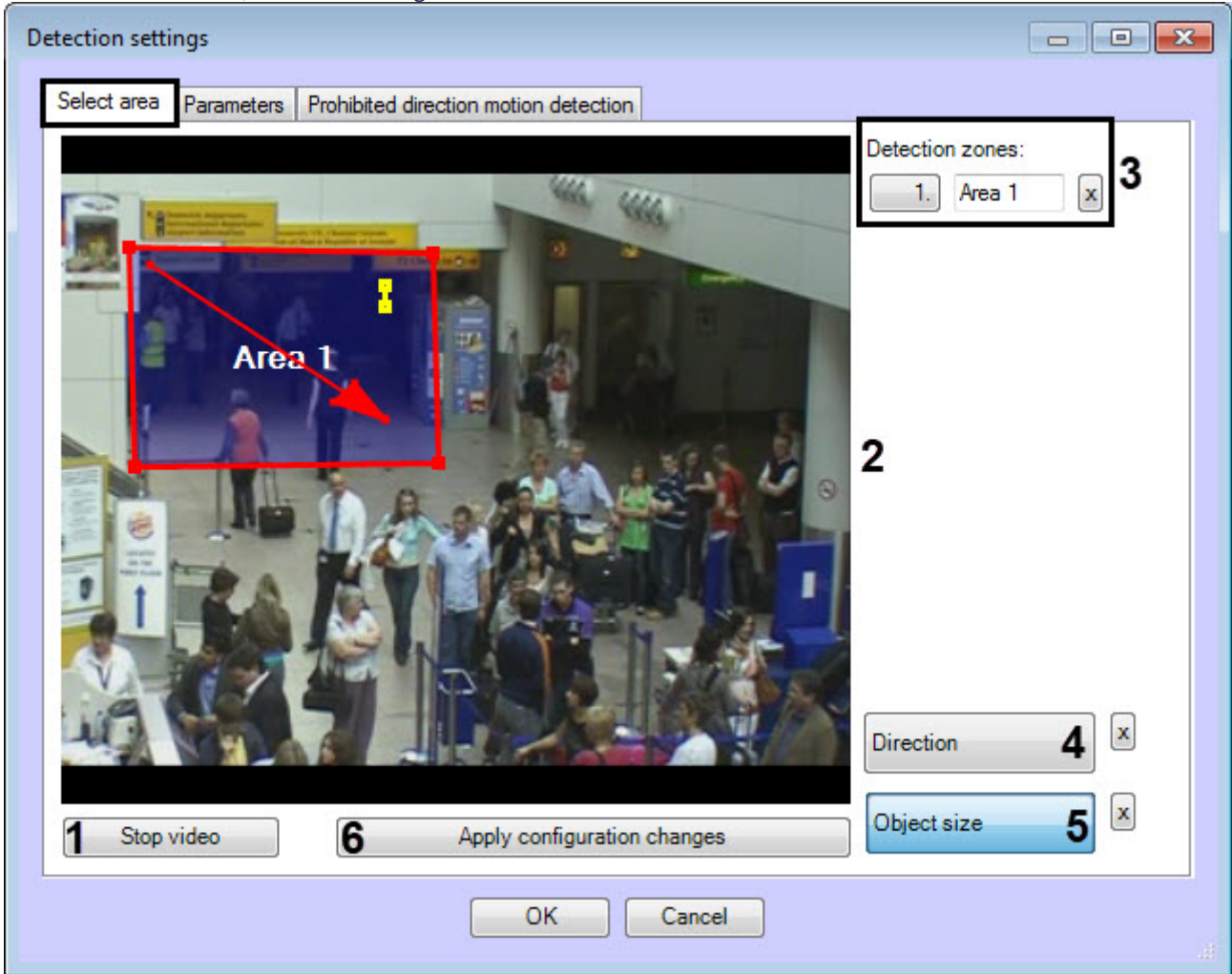
3. Click the **OK** button to complete the detection configuration.

The Videointellect camera state detection is now configured.

4.17.4.1.4 Videointellect prohibited direction motion detection

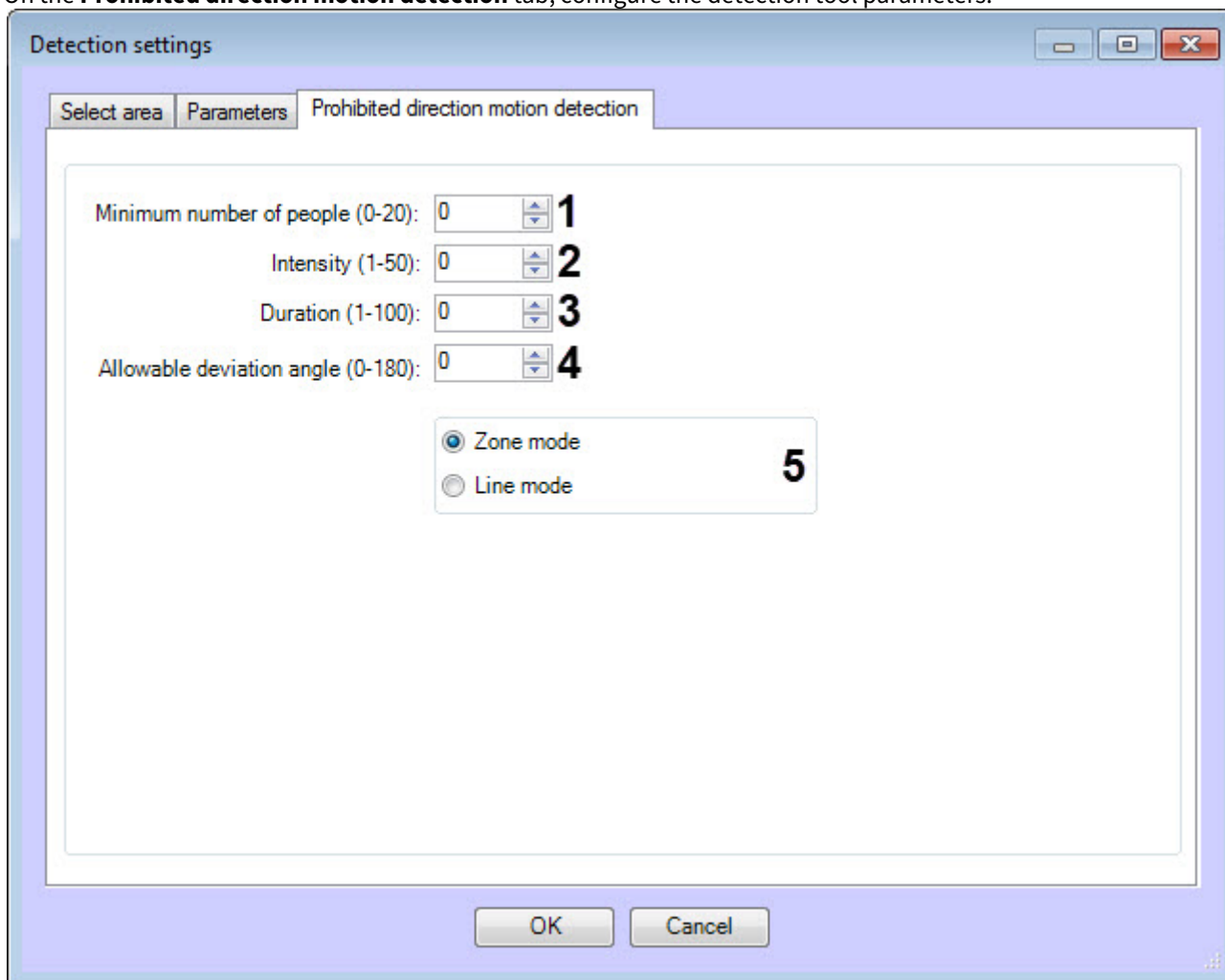
The Videointellect prohibited direction motion detection is configured as follows:

1. On the **Select area** tab, do the following:



- a. To capture a frame of a video image, click the **Stop video** button (1).
- b. Set one or more zones in which the motion will be detected (2).
- c. All set zones are displayed in the **Detection zones** area (3). If necessary, you can change the name of the zone or delete the zone.
- d. Click the **Direction** button (4) and use the arrow to set the correct motion direction. Moving against the direction of the arrow will be considered prohibited.
- e. Click the **Object size** button (5) and use the yellow line to set the approximate size of the human head.
- f. Click the **Apply configuration changes** button (6).

2. On the **Prohibited direction motion detection** tab, configure the detection tool parameters:

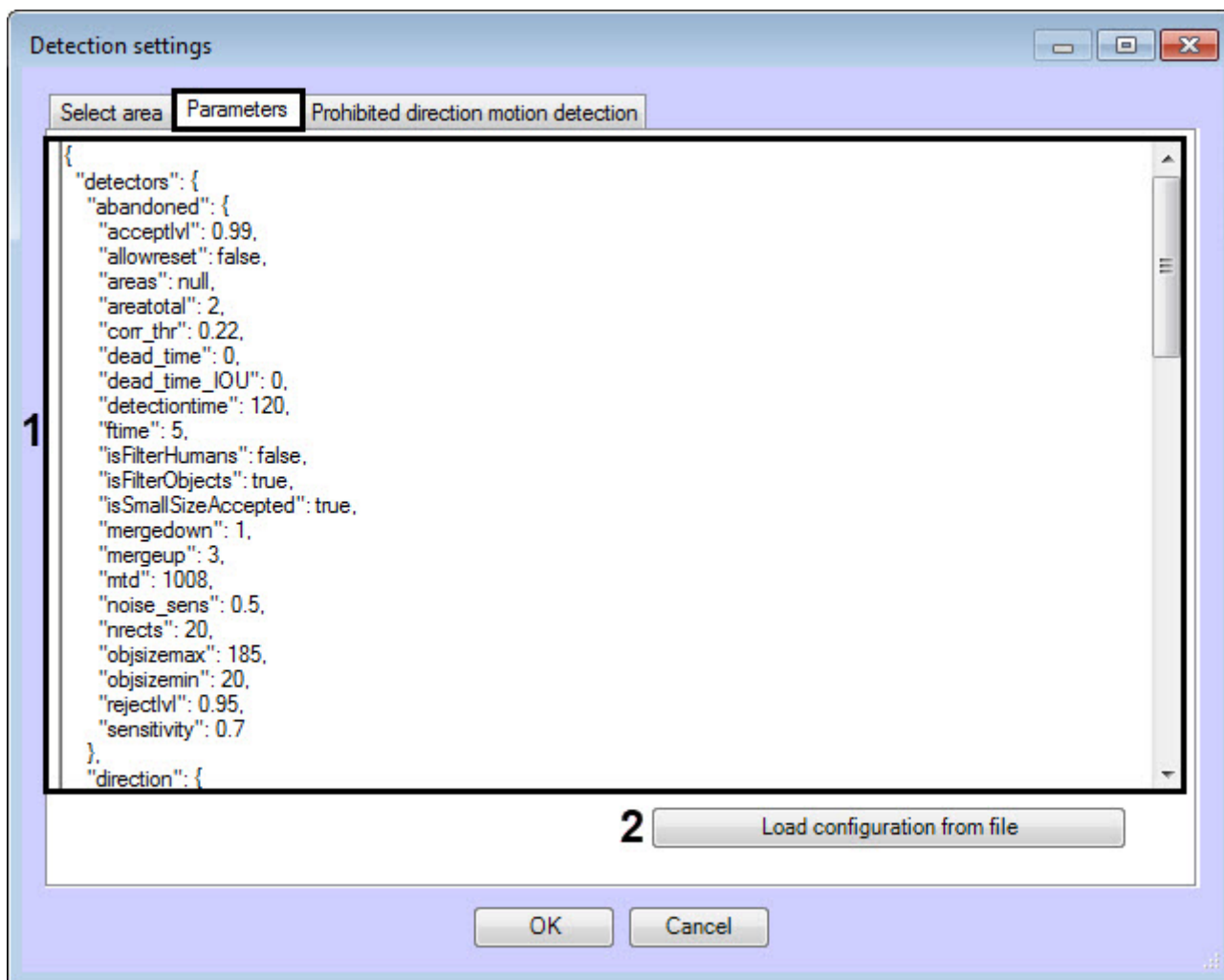


- a. In the **Minimum number of people (0-20)** field (1), enter the minimum number of people at which the detection tool starts recording an event when they cross a zone or line in a prohibited direction.
- b. In the **Intensity (1-50)** field (2), enter the sensitivity of the object's movement in the zone; when this value is exceeded, the detection tool is triggered.
- c. In the **Duration (1-100)** field (3), enter the duration in the object's movement conventional units in the prohibited direction.

Note

It can be used in the analysis of dense passenger traffic, in which an object making its way in the opposite direction appears in the camera's field of view only for short periods of time, and the rest of the time it is hidden from the camera's view by the crowd.

- d. In the **Allowable deviation angle (0-180)** field (4), enter the allowable angle in degrees, indicating the allowable spread of the movement direction of the object in the prohibited direction from the direction specified by the arrow (the so-called "detection cone").
 - e. Select the mode of motion in the prohibited direction:
 - **Zone mode** - movement in the zone against the direction of the arrow will be considered prohibited.
 - **Line mode** - crossing the line against the direction of the arrow will be considered prohibited.
3. On the **Parameters** tab, the area (1) displays the current detection configuration. This configuration can be copied to any text file. To upload a configuration from a file, click the **Load configuration from file** button (2).



Description of some configuration parameters:

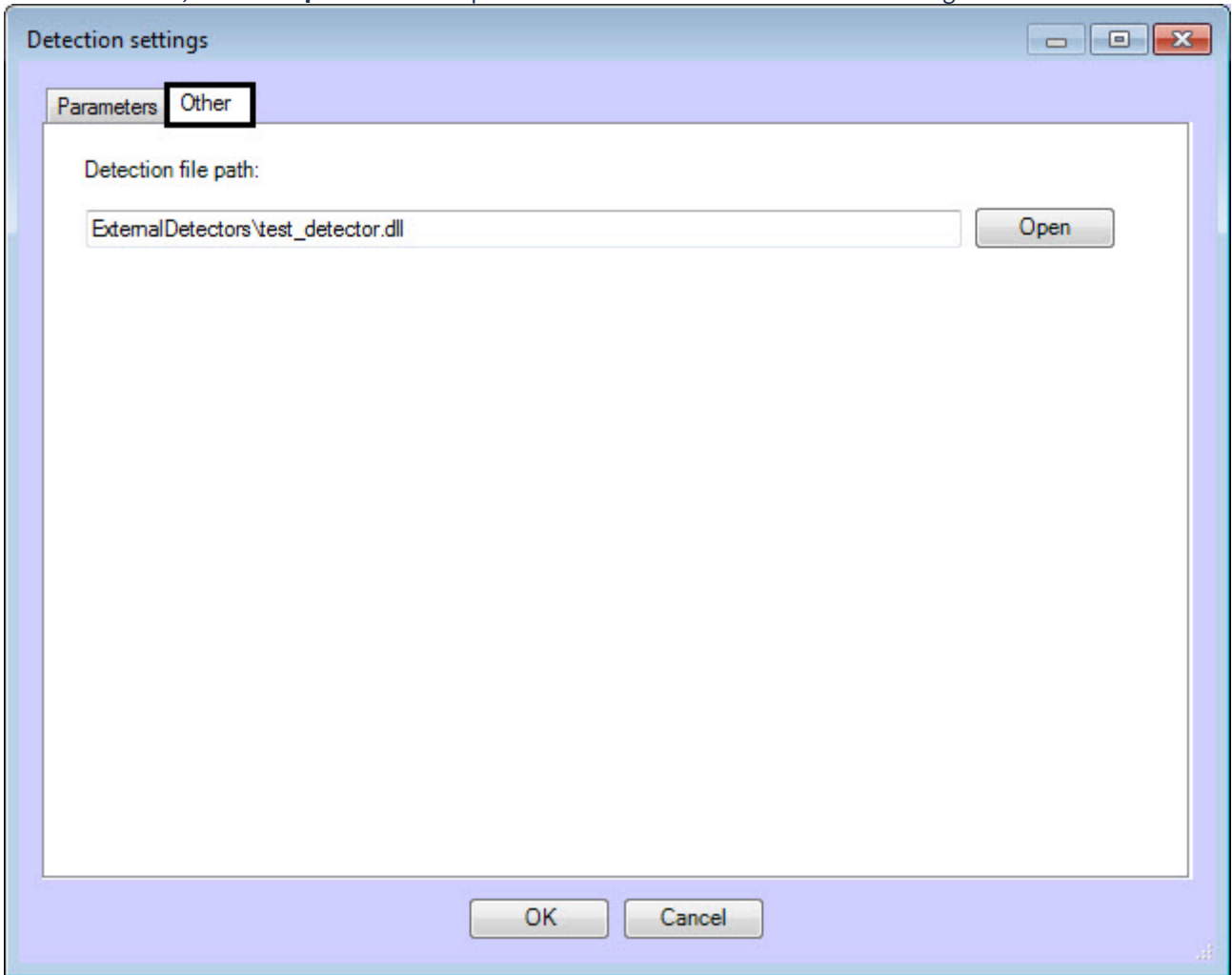
- a. minNumberOfObjects - Minimum number of people.
 - b. intensity - Intensity.
 - c. duration - Duration.
 - d. sizeobject - The size of the human head.
 - e. angle - Allowable deviation angle.
4. Click the **OK** button to complete the detection configuration.

The VideoIntellect prohibited direction motion detection is now configured.

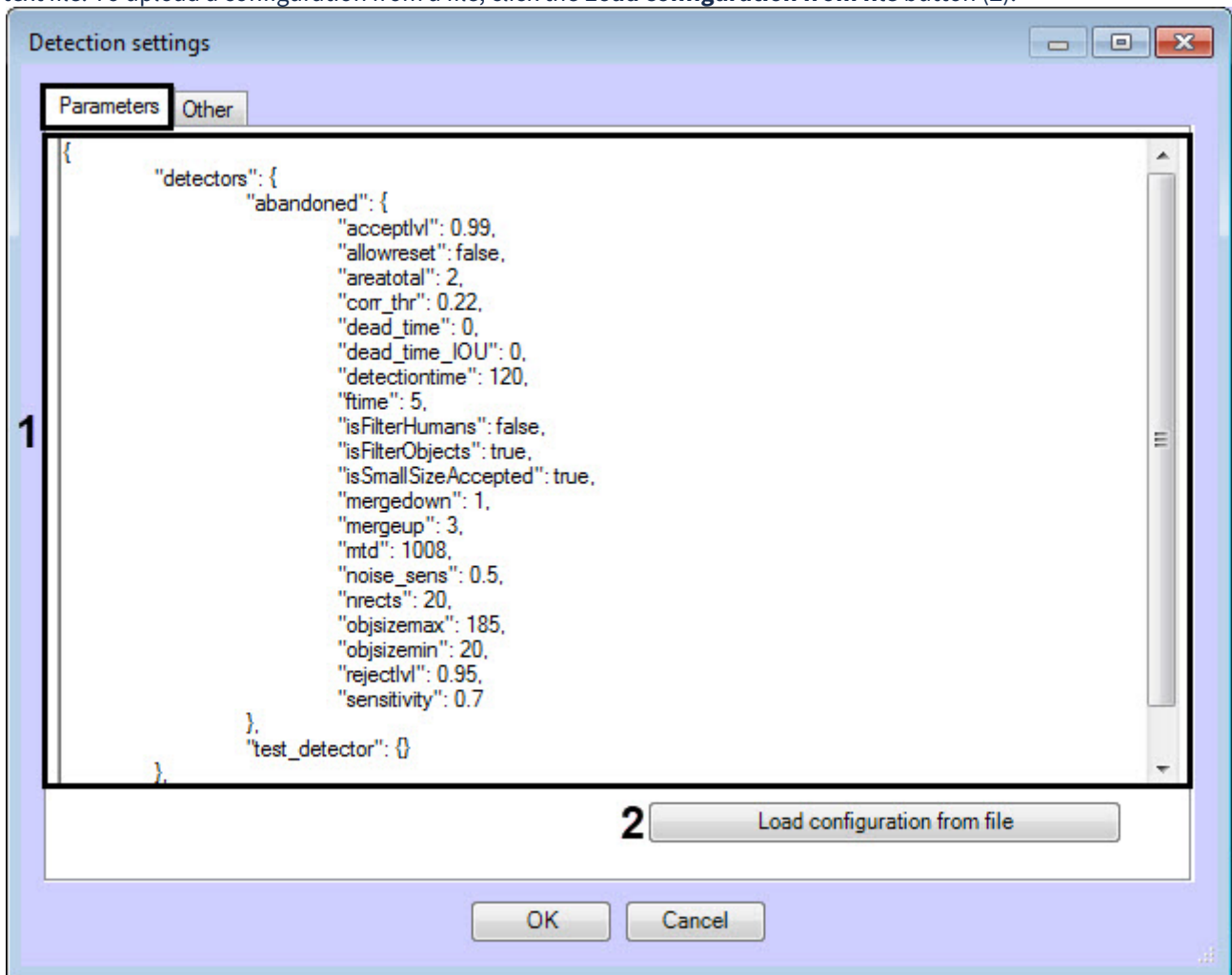
4.17.4.1.5 VideoIntellect other detection

The VideoIntellect other detection is configured as follows:

1. On the **Other** tab, click the **Open** button to open the .dll file of the VideoIntellect non-integrated detection tool.



- On the **Parameters** tab, the area (1) displays the current detection configuration. This configuration can be copied to any text file. To upload a configuration from a file, click the **Load configuration from file** button (2).



- Click the **OK** button to complete the detection configuration.

The Videointellect other detection is now configured.

4.18 Neurocounter

4.18.1 Functionality of the Neurocounter module

Neurocounter counts the number of objects in a given surveillance area using a neural network. When the event condition is met, an event with the number of detected objects is generated.

4.18.2 Video stream and scene requirements for Neurocounter module

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

Video stream from camera	<ul style="list-style-type: none"> The resolution is at least 640x360 pixels. It is also not recommended to use a resolution higher than 1920x1080, since higher resolution does not increase the detection quality, but significantly increases the consumption of resources. The optimal resolution for solving typical tasks is 1280x720 (see Example of configuring Neurocounter for solving typical task). The frame rate per second in the video stream from the camera is at least 8 for solving typical task. Both colorless (gray) and color images.
--------------------------	--

Lighting	<ul style="list-style-type: none"> • Lighting in the scene is at least 50 lux per square meter. In conditions of insufficient or excessive lighting (night or light-striking), stable operation of the video analytics is not guaranteed. • There are no abrupt changes in lighting.
Scene and camera angle	<ul style="list-style-type: none"> • Moving objects are visually separable from each other. • The background is mostly static and does not change abruptly. • Moving objects are minimally obscured by static objects in the scene (columns, trees, etc.). • The analyzed scene does not have reflective surfaces and sharp shadows from moving objects. If present, they should be masked. • Camera shake does not result in image offsets greater than 1% of the frame size.
Object images	Objects image requirements for Neurocounter

⚠ Attention!

Correct operation of the neural counter is not guaranteed when using a fish-eye lens.

✓ Software and hardware requirements


4.18.3 Objects image requirements for Neurocounter

To ensure the correct operation of **Neurocounter**, the following image requirements should be met:

1. The object to be detected is clearly distinguishable by the human eye.
2. The width or height of the objects does not exceed 75% of the frame size.
3. The image is not noisy and not distorted by compression algorithm artifacts.
4. The duration of the object's visibility is at least 6 frames.
5. The object moves in the certain direction between two adjacent frames at a distance which does not exceed the object's size. This condition is necessary for the correct calculation of the trajectory of the object (track).
6. The minimum value of pixel density per meter is observed:

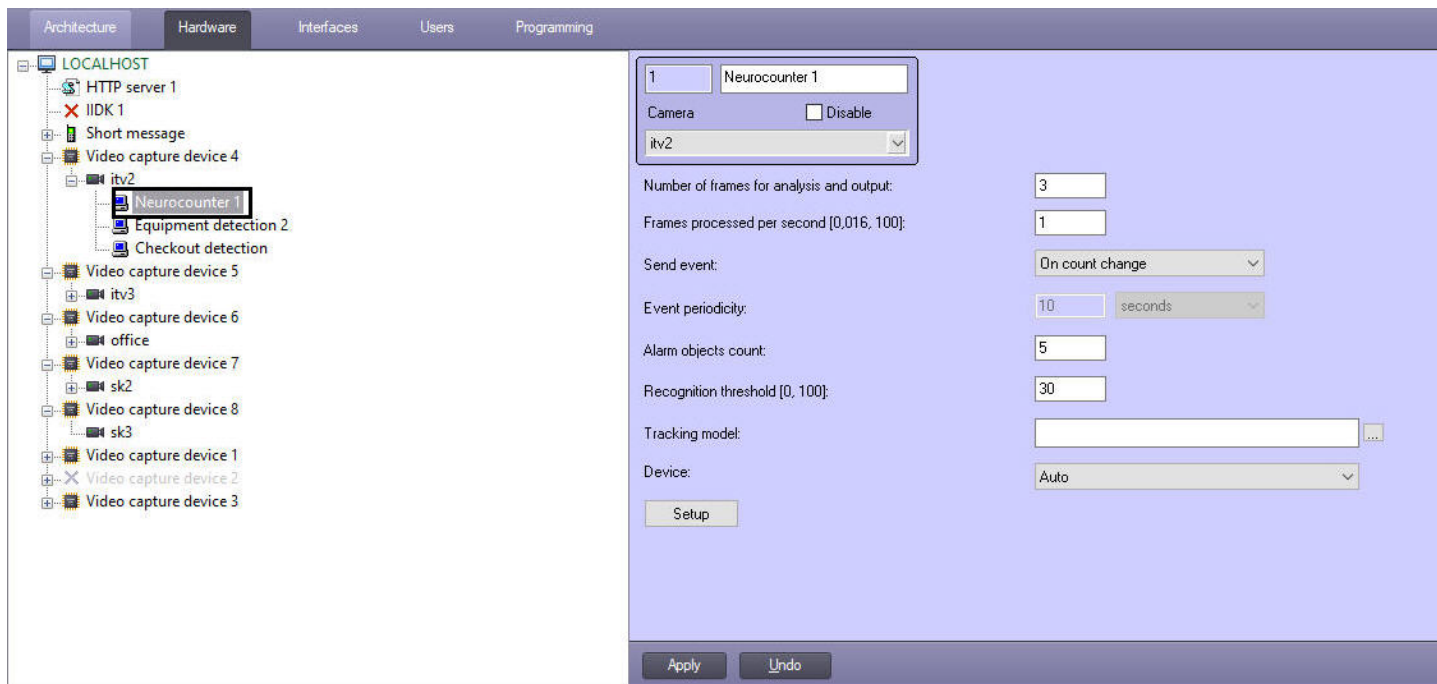
Image resolution	Object type	Minimum pixel density per meter (ratio of the object width in pixels to the object width in meters)	Minimum object size in pixels, Width x Height	Ratio of the object width to the frame width as a percentage
1920x1080	Human	55	~25x105	~3%
1280x720	Human	35	~17x70	~3%
640x360	Human	17	~10x42	~3%
1920x1080	Light vehicle (2 axles)	55	~354x300	~20%
1280x720	Light vehicle (2 axles)	35	~240x205	~20%
640x360	Light vehicle (2 axles)	17	~132x112	~2%

Image resolution	Object type	Minimum pixel density per meter (ratio of the object width in pixels to the object width in meters)	Minimum object size in pixels, Width x Height	Ratio of the object width to the frame width as a percentage
				0 %

 Video stream and scene requirements for Neurocounter module

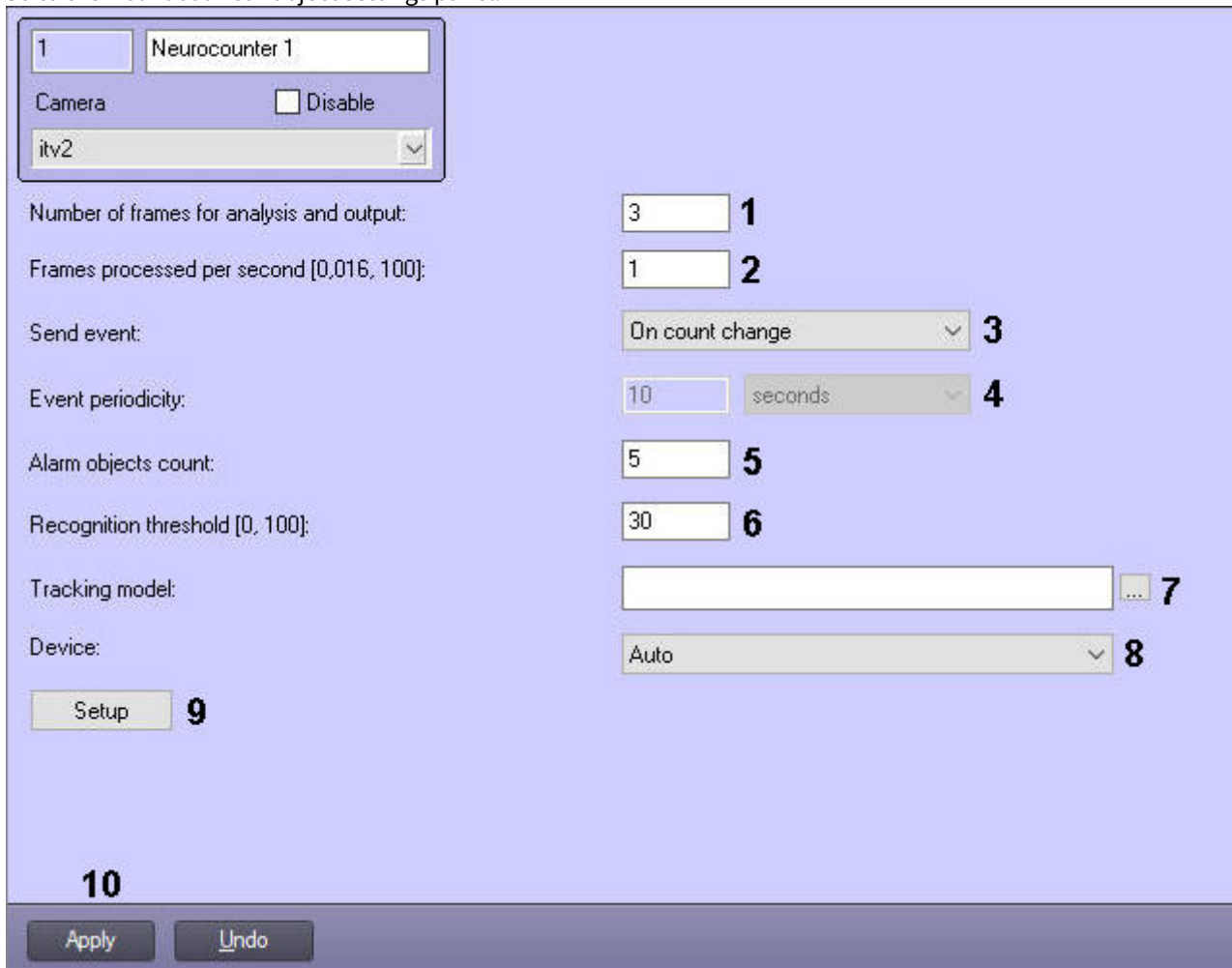
4.18.4 Configuring the Neurocounter module

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



The **Neurocounter** module is configured as follows:

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




2. In the **Number of frames for analysis and output** field (1), specify the number of frames to be processed to determine the number of objects on them.

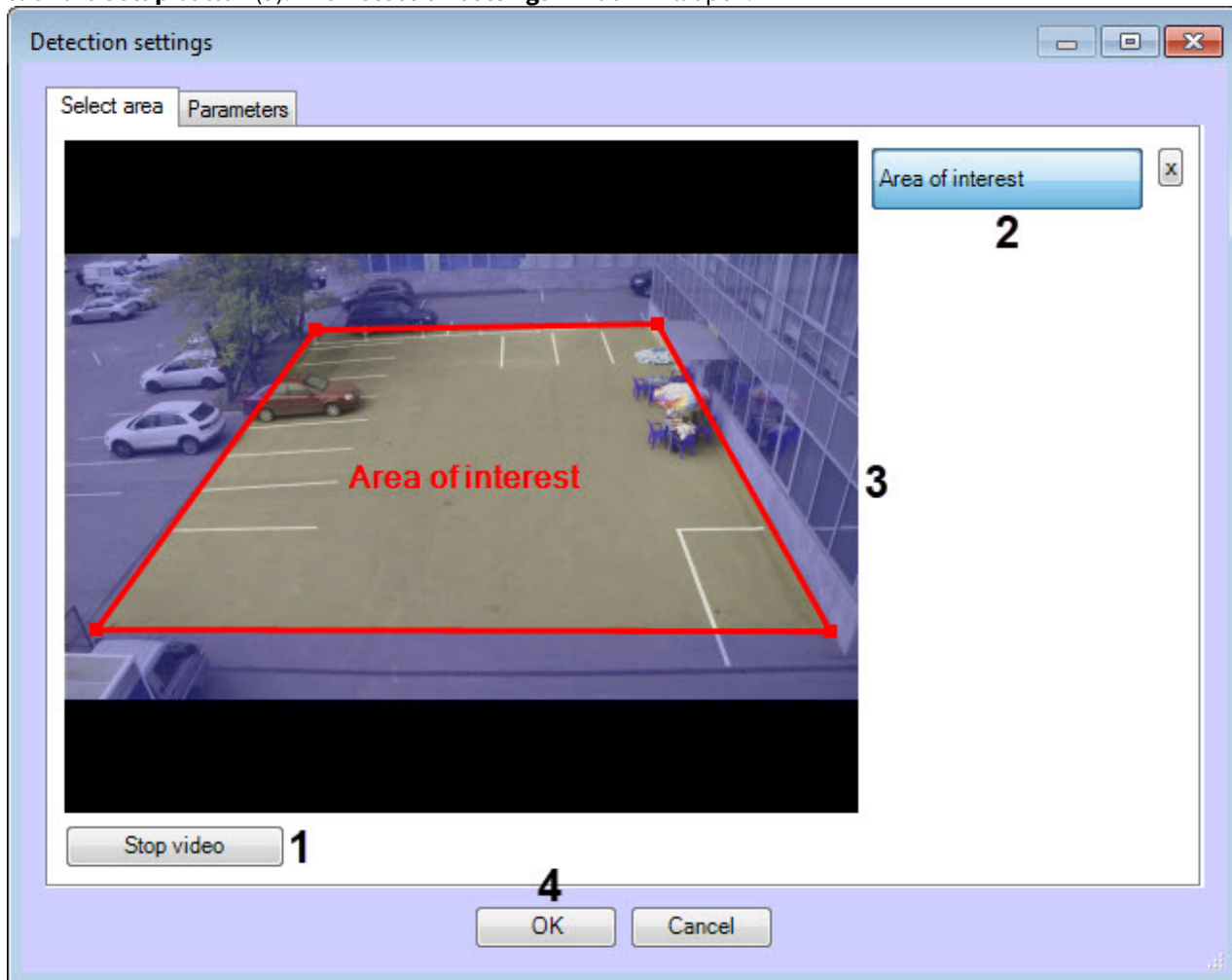
3. In the **Frames processed per second [0,016, 100]** field (2), set the number of frames per second that will be processed by the detection tool.
4. From the **Send event** drop-down list (3), select the condition by which an event with the number of detected objects will be generated:
 - **If threshold exceeded** - is triggered if the number of detected objects in the image is greater than or equal to the value specified in the **Alarm objects count** field.
 - **If threshold not reached** - is triggered if the number of detected objects in the image is less than or equal to the value specified in the **Alarm objects count** field.
 - **On count change** - is triggered every time the number of detected objects changes.
 - **By period** - is triggered by a time period:
 - i. From the **Event periodicity** drop-down list (4), select the time unit of the counter period: seconds, minutes, hours, days.
 - ii. In the **Event periodicity** field (4), set the time after which the event with the number of detected objects will be generated.
5. In the **Alarm objects count** field (5), set the threshold number of detected objects in the area of interest. It is used in the **If threshold exceeded** and **If threshold not reached** conditions.
6. In the **Recognition threshold [0, 100]** field (6), enter the neural counter sensitivity – integer value from **0** to **100**.

 **Note**

The neural counter 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.

7. Click the  button (7), and in the standard Windows box that opens, select the neural network file with the neural counter model.
8. In the **Device** drop-down list (8), select the device on which the neural network will operate.
9. Specify the detection surveillance area on the video image:

- a. Click the **Setup** button (9). 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).
10. Click the **Apply** button (10).

Configuring the **Neurocounter** module is complete.

4.18.5 Example of configuring Neurocounter for solving typical task

To detect objects with a speed less than 0.3 m/s, the following settings are recommended:

1. **The number of frames for analysis and output:** 3.
2. **The number of frames processed per second:** 1.
3. **Neural filter:** No.
4. **Recognition threshold:** 30.
5. **Neural network file:** Path to the *.ann neural network file. You can also select **Object type**.

Note

By default, the neural counter is configured for detection of objects with a speed less than 0.3 m/s.

To solve tasks in which the speed of the object is greater than 0.3 m/s, it is necessary to increase the number of processed frames and/or reduce the number of frames for analysis and output. The values are selected by trial-and-error method depending on the conditions of the task.

4.19 Equipment detection (PPE)

4.19.1 Functionality of the Equipment detection (PPE) module

The Equipment detection tool (PPE) locates individuals wearing no personal protective equipment within the area where it's required, and also individuals wearing improperly applied PPE.

⚠ Attention!

It is recommended to use the detector in a "gateway" environment: at the entrance to an area in which equipment or PPE is required, the employee is delayed for 5-10 seconds, during which the detection tool determines the presence of the necessary equipment (see [Examples of configuring Equipment detection tool \(PPE\) for solving typical tasks](#)).

For detection tool operation, at least two separate neural networks are used:

- Segmenting network — it structures up an image of a human body (locates head, shoulders, arms, hands, thighs, legs and feet);
- Classifying network — it detects equipment (PPE) on a specified body part, and checks if it's properly applied.

⚠ Attention!

To train a classification neural network, it is necessary to provide a list of equipment (see [Example of providing a list of valid equipment at the facility](#)).

4.19.2 Video stream and scene requirements for Equipment detection tool (PPE) operation

The requirements for cameras that will work with the **Equipment detection (PPE)** module are shown in the following table.

Video stream from camera	<ul style="list-style-type: none"> • The resolution is at least 640x360 pixels. It is also not recommended to use a resolution higher than 1920x1080, since higher resolution does not increase the detection quality, but significantly increases the consumption of resources. The optimal resolution for solving typical tasks is 1280x720. • The frame rate per second in the video stream from the camera is at least 3 for solving typical tasks. • Color image only.
Lighting	<ul style="list-style-type: none"> • Lighting in the scene is at least 200 lux per square meter. In conditions of insufficient or excessive lighting (night or light-striking), stable operation of the video analytics is not guaranteed. • There are no abrupt changes in lighting.
Scene and camera angle	<ul style="list-style-type: none"> • Moving objects are visually separable from each other. • The background is mostly static and does not change abruptly. • There are no products made of rods in the detection area. • Moving objects are minimally obscured by static objects in the scene (columns, trees, etc.). • The analyzed scene does not have reflective surfaces and sharp shadows from moving objects. If present, they should be masked. • Camera shake does not result in image offsets greater than 1% of the frame size.
Object images	Objects image requirements for Equipment detection tool (PPE)

⚠ Attention!

Correct operation of the detection tool is not guaranteed when using a fish-eye lens.

✔ Software and hardware requirements

4.19.3 Objects image requirements for Equipment detection tool (PPE)

To ensure the correct recognition of personal protective equipment, the following image requirements should be met:

1. The object to be detected (PPE) is clearly distinguishable by the human eye.
2. The image is not noisy and not distorted by compression algorithm artifacts.
3. The width or height of the equipment does not exceed 75% of the frame size.
4. The duration of the object's visibility is at least 3-8 frames. The minimum number of frames depends on the task.
5. There are no visible physical barriers between the camera lens and the analyzed object.
6. The minimum value of pixel density per meter is observed:

Image resolution	Object type	Minimum pixel density per meter (ratio of the object width in pixels to the object width in meters)	Minimum object size in pixels, Width x Height
1920x1080	Human	170	~102x309
1280x720	Human	128	~77x233
640x360	Human	80	~48x145

7. The minimum dimensions of the equipment on the body areas in pixels are observed. An example of equipment dimensions for a resolution of 1920x1080:
 - a. Upper body (torso) 75*100.
 - b. Legs 75*105.
 - c. Head 60*65.
 - d. Hands 65*60.
 - e. Feet 45*40.
 - f. Set of equipment 165*295.

✔ Video stream and scene requirements for Equipment detection tool (PPE) operation

4.19.4 Example of providing a list of valid equipment at the facility

✔ Data collection requirements for neural network training

For correct detection, it is important to understand exactly what items of equipment are used at the facility.

A complete list of equipment should be provided regardless of the current season of the year. This will help to reduce the number of false events from the Equipment detection tool (PPE) and get the most positive experience from using this analytics.

Below are examples of lists in the form of a table:

Gloves #1



Gloves #2



Gloves #3



Gloves #4

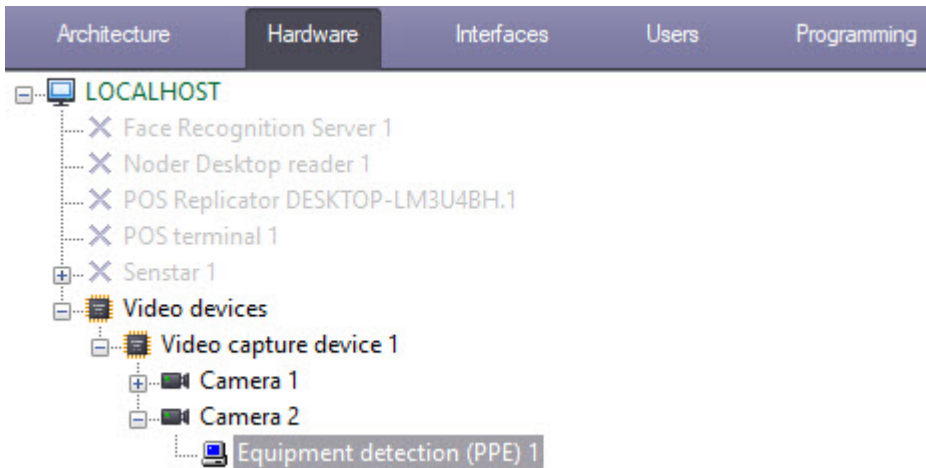


Helmet #1:			
			
Helmet #2:			
			
Helmet #3:			
			
Helmet #4:			
			
Helmet #5 (#4 screen):			
			
Helmet #6 (#4 earmuffs):			
			
<div style="display: flex; justify-content: space-between; align-items: center; background-color: #333; color: white; padding: 5px;"> ← UpperBody+Shoes Helmets Gloves + </div>			



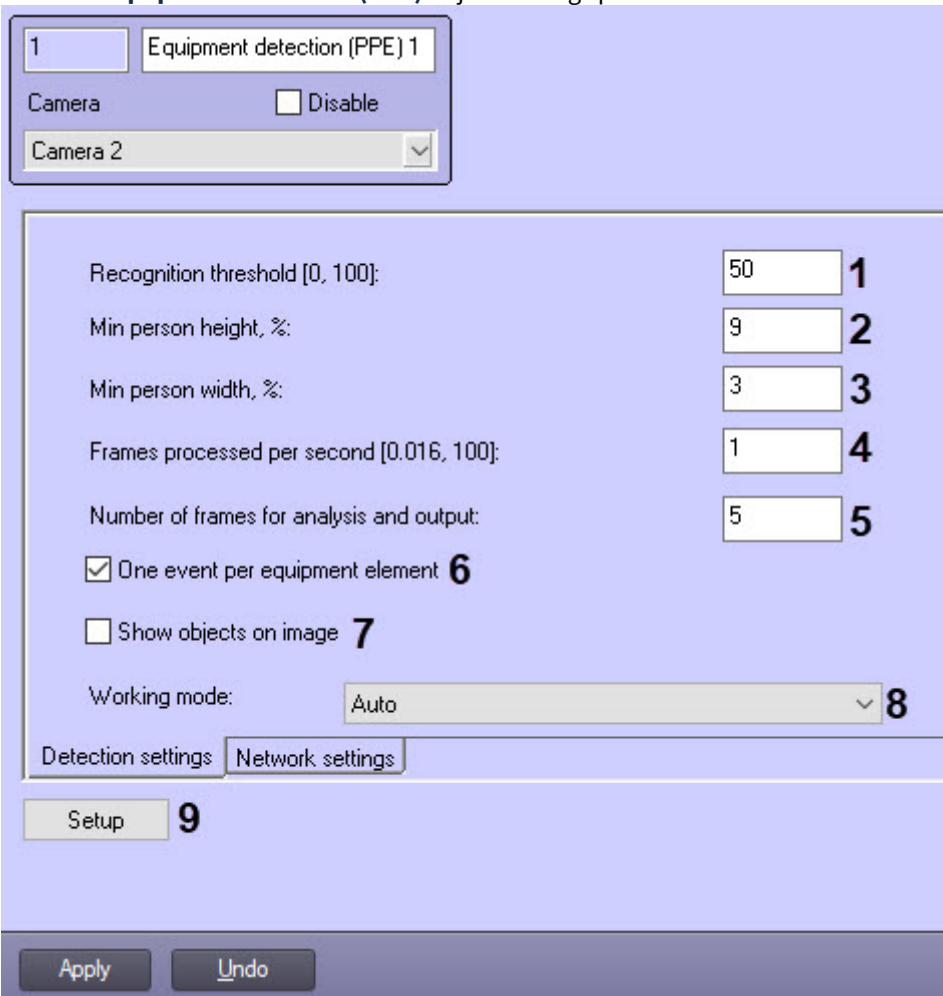
4.19.5 Configuring the Equipment detection (PPE) module

The **Equipment detection (PPE)** module is configured on the settings panel of the **Equipment detection (PPE)** object created on the basis of the **Camera** object on the **Hardware** tab of the **System settings** dialog window.



The **Equipment detection (PPE)** module is configured as follows:

1. Go to the **Equipment detection (PPE)** object settings panel.



2. In the **Recognition threshold [0, 100]** field (1), enter the detection tool sensitivity - an integer value from 0 to 100.

Note

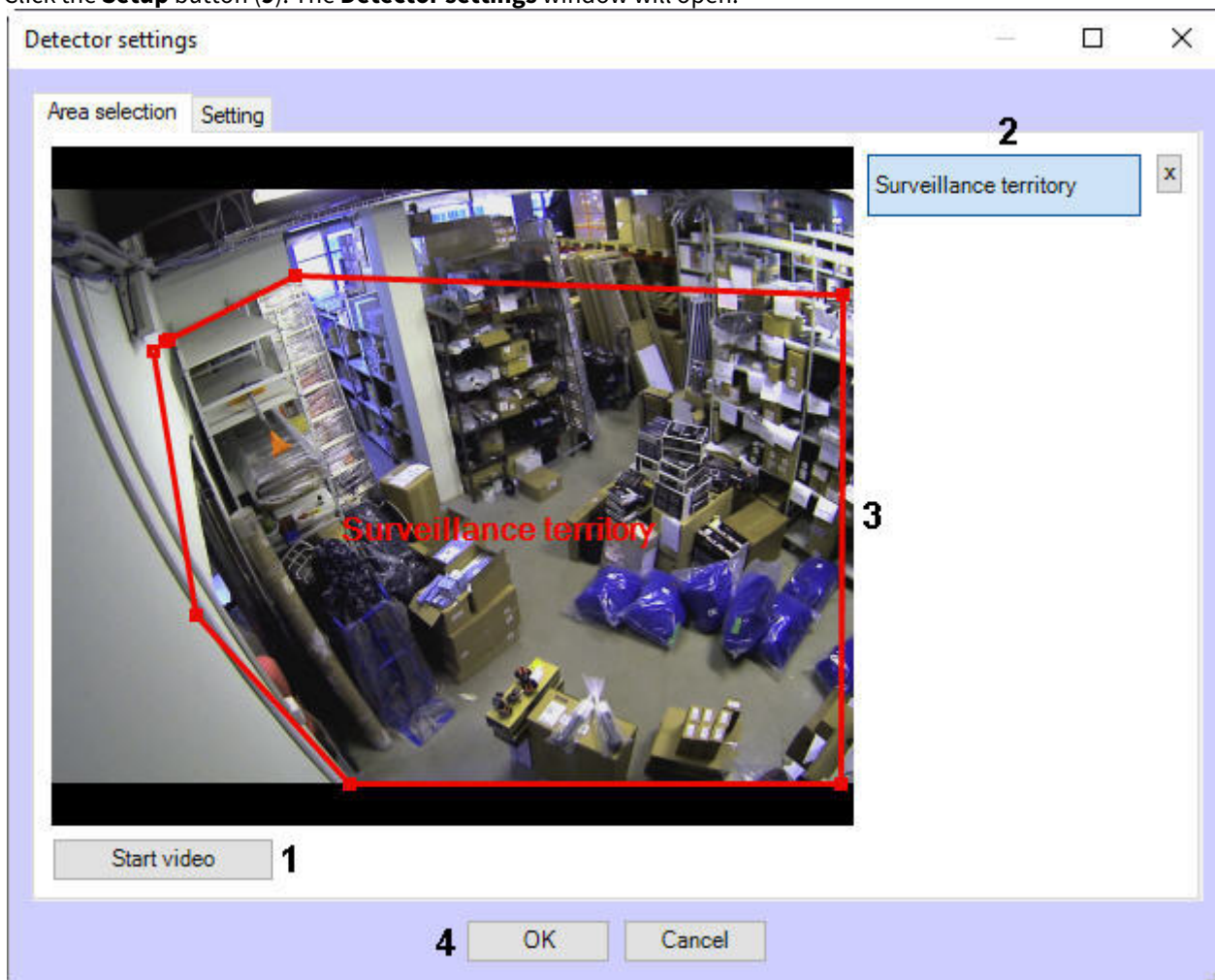
The detector sensitivity is determined experimentally. The lower the sensitivity, the greater the probability of false positives. The higher the sensitivity, the less chance of false alarms, however, some useful tracks may be skipped.

3. In the **Min person height, % (2)** and **Min person width, % (3)** fields, enter the minimum height and width of a person in the frame as a percentage of the frame height/width. Objects smaller than the specified size will not be detected.
4. In the **Frames processed per second [0,016, 100] field (4)**, set the number of frames per second that will be processed by the detection tool.
5. In the **Number of frames for analysis and output field (5)**, enter the minimum number of frames on which a violation should be detected in order to generate a trigger. The value should be in the [2; 20] range.
6. By default, the detection triggering is generated for each equipment violation. If it is necessary to generate a trigger once for each equipment item violation within a whole person's track, then uncheck **One event per equipment element** checkbox (6).

Note

Example. A person appeared in the frame without a helmet, then put it on and then took it off again. If the **One event per equipment element** checkbox is set, then there will be one trigger, if not – two.

7. Set the **Show objects in image** checkbox (7) if it is necessary to highlight the detected object with a frame on the image in the **Monitor** interface object window.
8. From the **Working mode** drop-down list (8) select the device on which the neural network will operate.
9. Click the **Setup** button (9). The **Detector settings** window will open.



- a. Click the **Start video** button (1) to capture a frame of the video image.
- b. Click the **Surveillance territory** button (2).
- c. On the captured video frame, sequentially set the nodal points of the area in which the objects will be detected by left-clicking the mouse button (3).
- d. Click **OK** (4).

10. Go to the **Network settings** tab (10).

The screenshot shows the configuration interface for the Equipment detection (PPE) module. At the top, there is a section for 'Equipment detection (PPE) 1' with a 'Camera' dropdown menu set to 'Camera 1' and a 'Disable' checkbox. Below this is the 'Network settings' tab (10), which contains a 'Segmenting network file' field (11) and five 'Classification network' file fields (12). A 'Setup' button is located below the file fields. At the bottom of the interface, there are 'Apply' (13) and 'Undo' buttons.

11. Select the segmenting neural network file (11).
12. Select one or several files of the classification neural network (12). Each classification neural network detects equipment on a specific body segment.
13. Click the **Apply** button (13) to save the changes.

The **Equipment detection (PPE)** module is now configured.

 [Examples of configuring Equipment detection tool \(PPE\) for solving typical tasks](#)

4.19.6 Examples of configuring Equipment detection tool (PPE) for solving typical tasks

Typical tasks for detecting personal protective equipment are the following:

1. Detection in gateway conditions. A gateway is a border, which can be either a virtual line or a door, a barrier, a turnstile. The algorithm for working in gateway conditions is as follows:
 - a. A person stops in front of an area where PPE is required.
 - b. The person poses in a way that it is possible to check the presence of all items of equipment (the item is not overlapped by the person himself, other items of equipment, items of clothing).
 - c. Human screening. There should be no obstacles between the person and the camera, blocking the person for screening.
2. Detection in a production conditions: persons move freely in the detection area.

 **Note**

By default, the Equipment detection tool (PPE) is configured for detection in gateway conditions.

The recommended settings for solving typical tasks are as follows:

Settings	Equipment detection in gateway conditions	Equipment detection in production conditions
The number of frames processed per second	1	3
Minimum person height	0,01	0,09
Minimum person width	0,01	0,03
Number of frames for analysis and output	3	7
One event per item of equipment	Yes	Yes

4.20 Traffic light detection

4.20.1 Functionality of the Traffic light detection module

The **Traffic light detection** module is designed for the following functions:

1. Determining the traffic light state without connection to the traffic light controller.
2. Creating the messages about permitted and forbidden moving directions.
3. Transmitting the data about permitted and forbidden moving directions to the **Traffic violations detection** module.

 **Note.**

The **Traffic light detection** module is used in the **Traffic violations detection** software module, which is part of *Auto Intellect* (see [Traffic violations detection](#)).

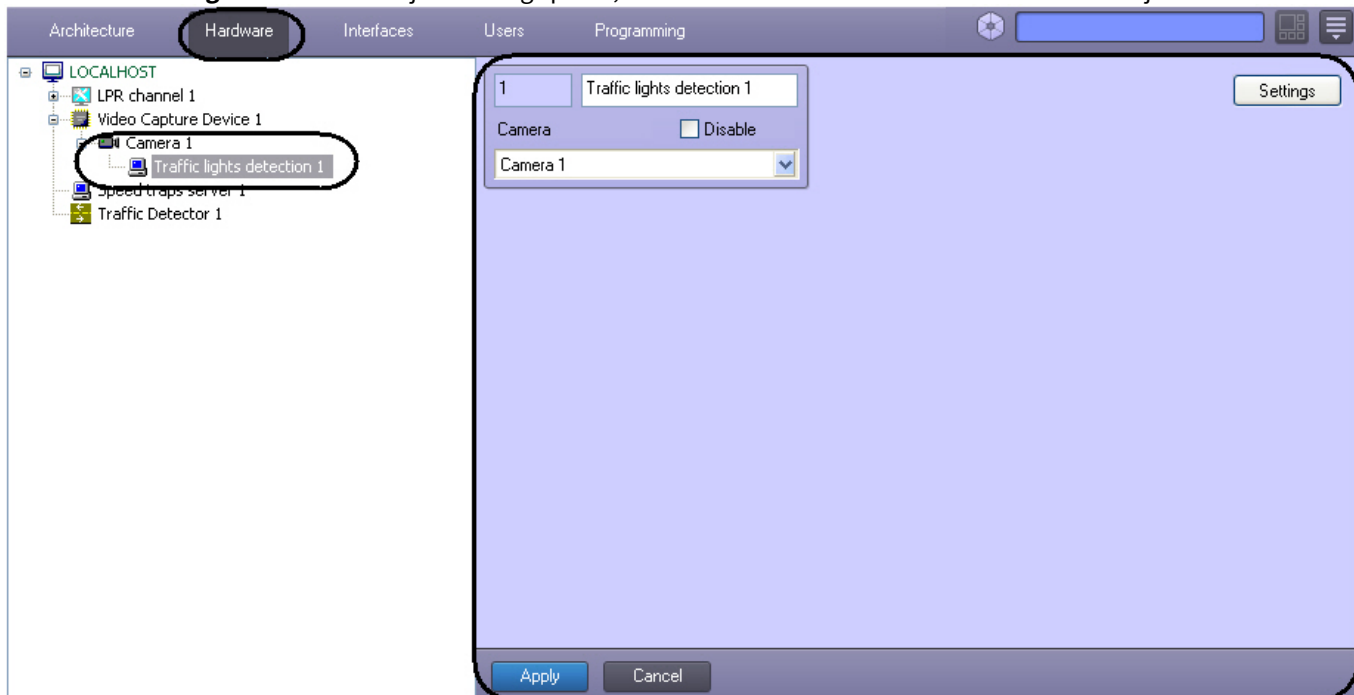
4.20.2 Configuring the Traffic light detection module

 **Note.**

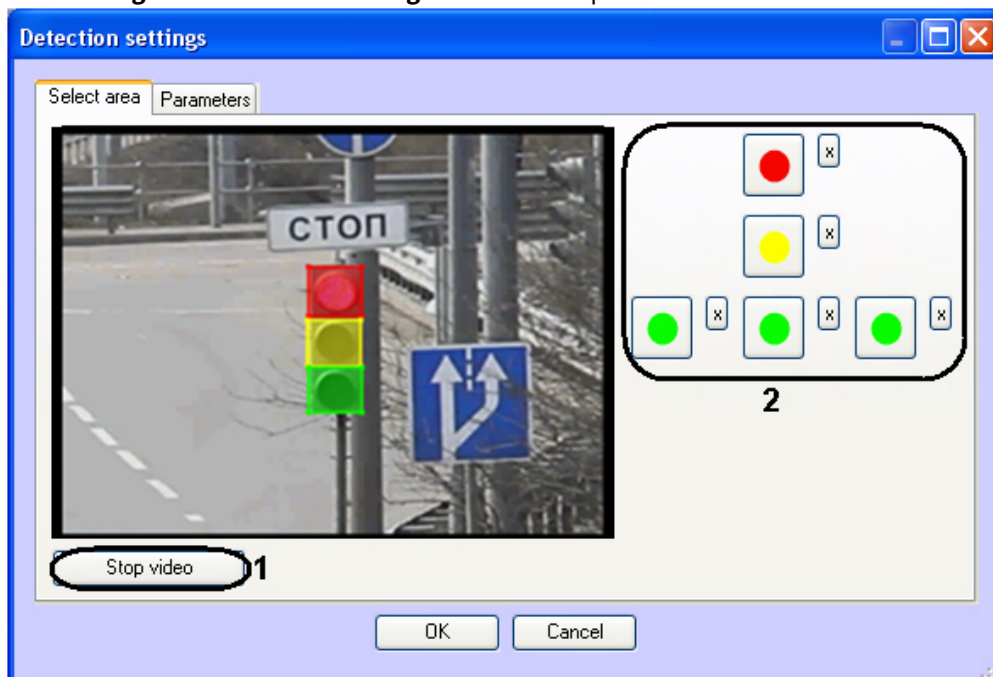
For proper working of the **Traffic lights detection** module traffic light is to be always observed for camera and it is not to be covered by some items or other vehicles.

To setup the **Traffic lights detection**, do the following:

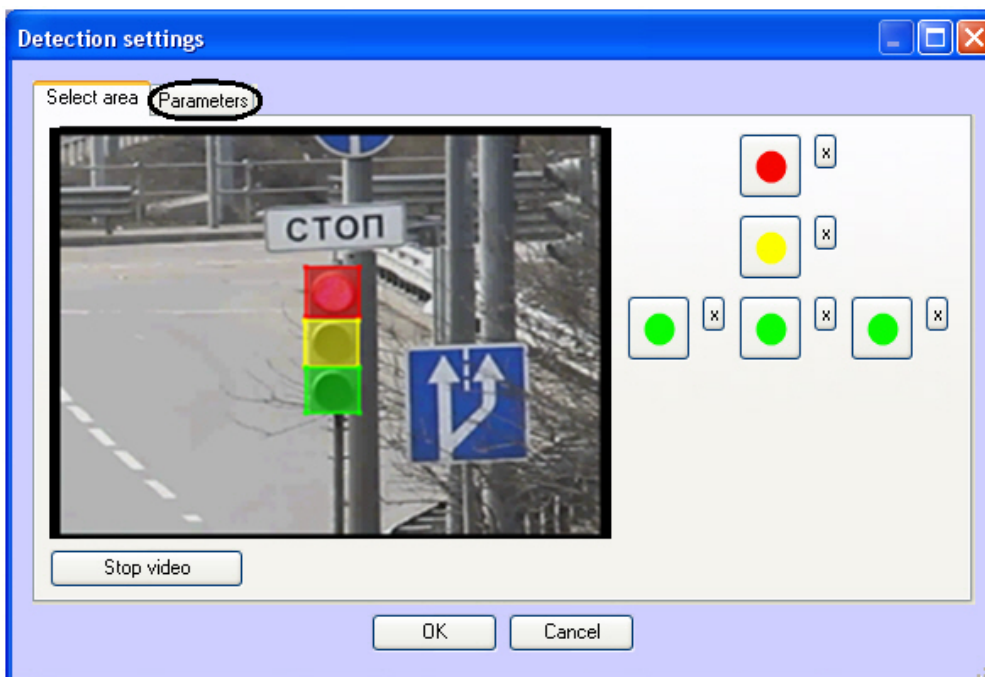
1. Go to the **Traffic lights detection** object settings panel, which is created on the basis of the **Camera** object.




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



3. Specify the area of traffic light signals:
 - a. Click **Stop video** to capture the video frame (1).
 - b. Do the following actions for each traffic light signal:
 - click the button with this signal (2);
 - specify the area of this signal on the captured video frame. Click the left mouse button on the video frame and move the cursor holding it.

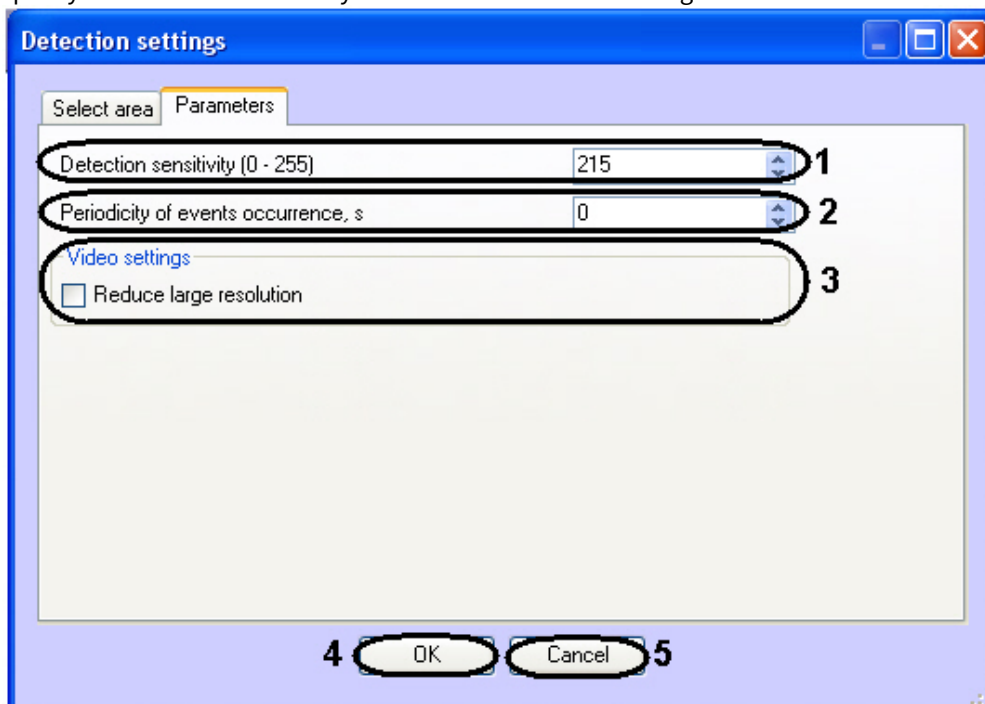


Note.

To remove the area click  opposite the corresponding button.

4. Specify the software module parameters:

- a. Go to the **Parameters** tab.
- b. Specify the detection sensitivity in conventional units in a range from 0 to 255 in the corresponding field (1).



- c. In the **Periodicity of events occurrence, s** field (2), enter the time period (in seconds) between creating messages about the states of traffic light signal.
 - d. Set the **Reduce large resolution** checkbox if the state of traffic light signal is to be recognized on the frame (3).
5. Click **OK** to save changes and to return to the settings panel of the **Traffic Light Detection** object (4).

**Note.**

Click **Cancel** to return to the settings panel of **Traffic Light Detection** object without saving changes (5).

6. Click **Apply**.

Setting up the **Traffic Light Detection** software module is completed.

5 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](#).

5.1 Operating the «Queue length detection» module

5.1.1 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](#).

5.1.2 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](#).

5.1.3 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](#).



5.2 Operating the «People counter detection» module

5.2.1 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](#).

5.2.2 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](#).

5.2.3 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.](#)



5.3 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](#).

5.4 Operating the Detection of light indication control module

The **Detection of light indication control** module sends messages to the **Event viewer** when the light source is detected or lost in the surveillance area.

Source	Event	Region	Add. info	Card	Date and time
Detection of light indication control 1	Enabled		2		01.07.2021 16:20:26
Detection of light indication control 1	Disabled		2		01.07.2021 16:20:27

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.

5.5 Operating the «Heat map detection»

5.5.1 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](#).

5.6 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](#).

5.7 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":0.99811}]}		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](#)).

5.7.1 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](#)).

5.8 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.

5.9 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.

5.10 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](#).

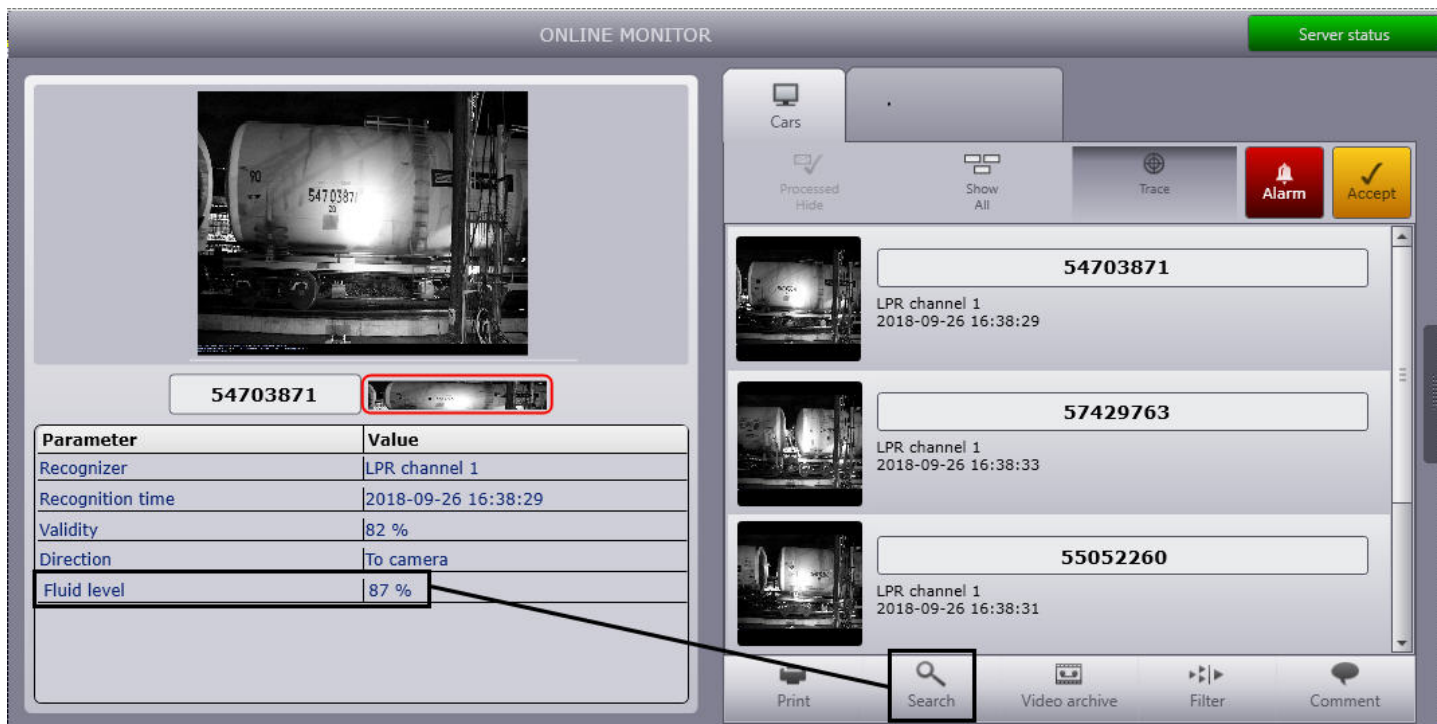
Event viewer 1						<input type="checkbox"/> Show filters	Clear
Source	Event	Region	Add. info	Card	Date and time		
● Fire detection 1	Fire detected				04.10.2017 13:00:52		
● Fire detection 1	Fire stopped				04.10.2017 13:01:00		

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](#).

5.11 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
}

```

5.12 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](#).

When you use the the neurotracker track counter, the **Track counter** object will receive the **Number of tracks** events that contain the number of recorded objects. The counting of the number of objects starts from 0, not from 1, i.e.: 0, 1, 2, 3, etc.

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](#).

5.13 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](#).

5.14 Operating the VideoIntellect embedded detector

In case the alarm event is detected in the surveillance area, the corresponding VideoIntellect detection tool sends the message from the object to the **Event viewer** interface window. For details, see [Configuring the VideoIntellect embedded detector module](#).

For details on 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](#).

5.15 Operating the Neurocounter module


In case the objects are detected in the monitored area, the **Neurocounter** sends the events with the corresponding number of objects to the **Event viewer** interface window.

6 Appendix 1. Debug window

6.1 General information

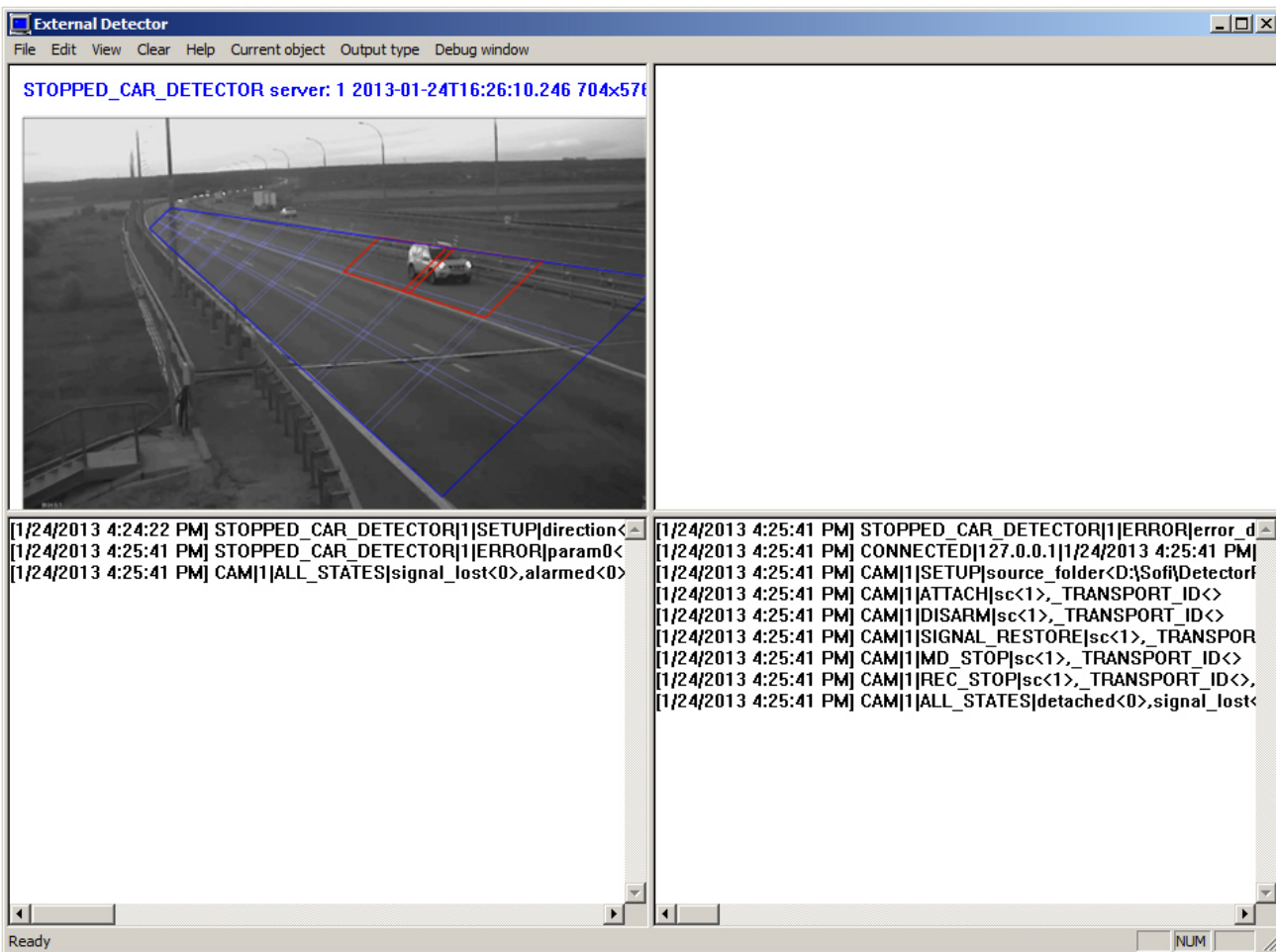
The debug window is designed to control events received from detections registered in the system. Besides, the function of displaying the detection area above the video image received from camera is available in the debug window.

6.2 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.

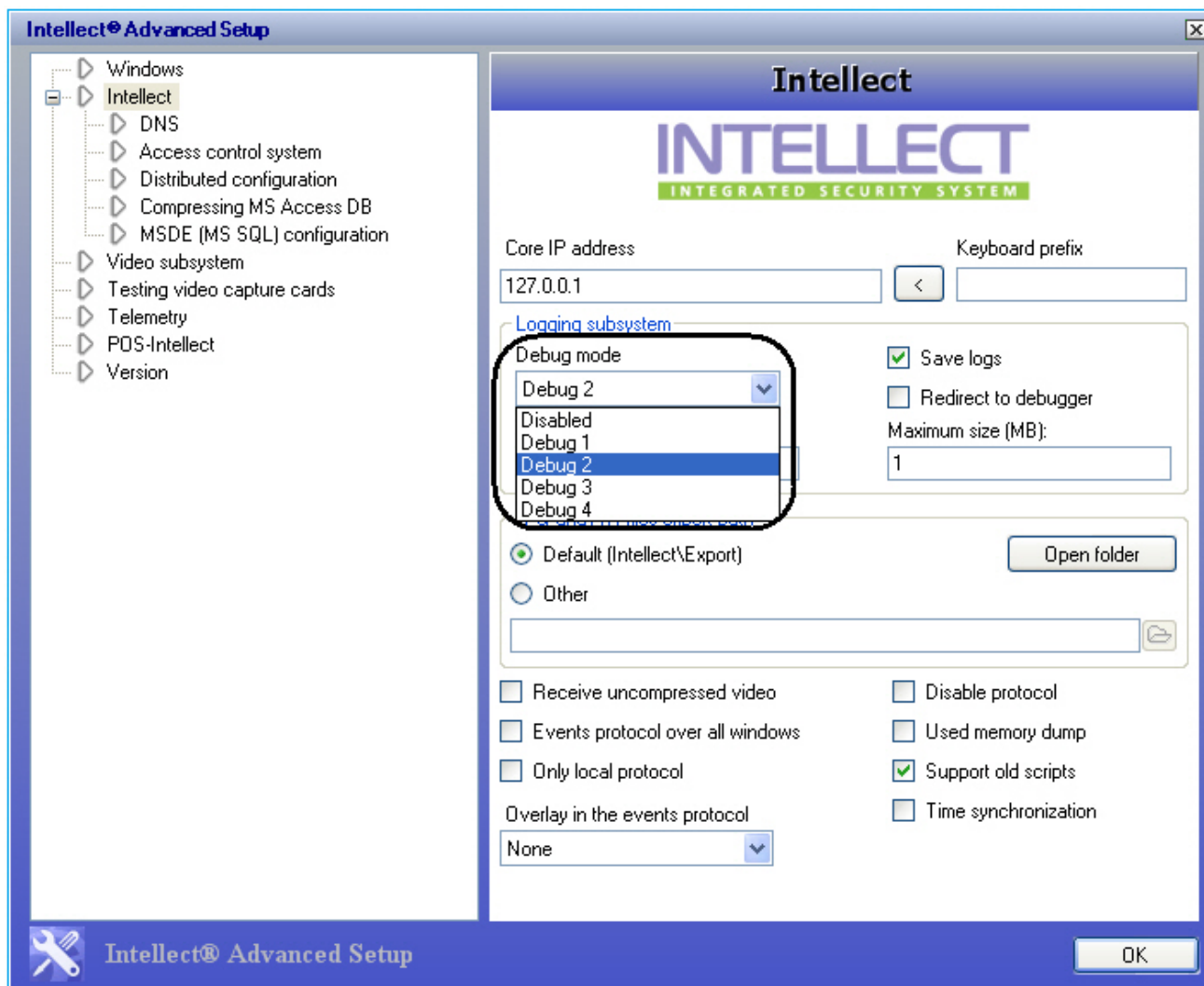


As a result the External detector window is displayed.



Attention!

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



6.3 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"> the person size for queue length detection and people counter detection; detection zones for stopped vehicle detection.
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	File menu	Access to the Exit function.
5	Edit menu	Access to the operations with text.
6	View menu	Access to the function of displaying and hiding the status bar.
7	Clear button	Clear areas of events viewing.
8	Help menu	Access to the information about program.

9	Current object 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.
---	----------------------------	--

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

7.1 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, <> - y coordinates
x.0<>...	0 - serial number of a track (object) in the event, <> - x coordinates
id.0<>...	0 - serial number of a track (object) in the event, <> - object id
id.count<>	Current number of tracks (objects)

7.2 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);
```