engiAI Ethernet System

Quick Start

How to connect the camera, create an AI model, and start inspection in just five steps

Step 1 Camera	**Step 2** Model and classes	**Step 3** Zones and images	**Step 4** Training	**Step 5** Inspection

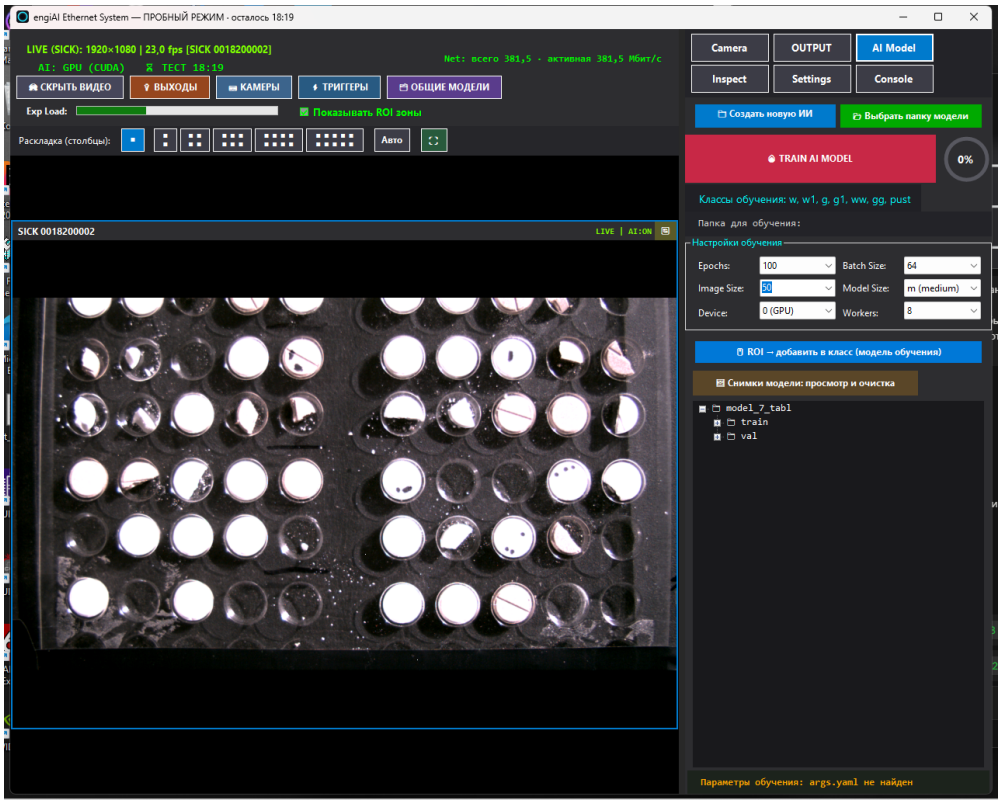
All screenshots in the guide were taken with a working program on a real SICK picoCam camera. The program also works with HIKROBOT (GigE), USB, and RTSP cameras.

Contents

How to use the manual. Sections 1-5 guide you from a blank program to a working inspection; complete them in order, as each takes a few minutes. Section 6 shows how to expand the model with a new class. Sections 7-8 explain what to do after the program has started: rejector, triggers, general models, and a short checklist.

Program Interface in One Minute

The main window is divided into two sections. **On the left**, you'll find a live video feed from the cameras, with a status bar above it. **On the right**, there's a panel with tabs, which contains all the controls. You can drag the border between the two sections with your mouse; the program remembers the size and position of all the windows.



Main window: video camera on the left, AI Model tab on the right.

Buttons above the video

Button	What it opens
HIDE VIDEO	Removes the image from the screen. Analysis continues – only the drawing process is saved.
OUTPUTS	Reference window: contacts of discrete camera outputs, testing with a multimeter, connection loads.
CAMERAS	Adding and removing cameras, and viewing their settings.
TRIGGERS	Connecting an external sensor to the camera's discrete input.
GENERAL MODELS	Sets of models assigned to multiple cameras simultaneously.
Layout (columns)	How many camera tiles should be displayed in a row; the green button to the right of "Auto" opens Full-screen cameras.

Tabs on the right

Tab	Purpose
Camera	Parameters of the selected camera: shutter speed, gain, color, resolution, and automatic settings for AI.
OUTPUT	rejector: rejection rules – TCP packets to the robot/PLC and discrete outputs from the camera.
AI Model	Creating the model: folder, classes, dataset, and training.
Inspect	Active tab: Which model is performing the inspection, the inspection areas, and what signal should be used? Shot.
Settings	Language of the interface, purchasing a license, saving/loading settings, video recording.

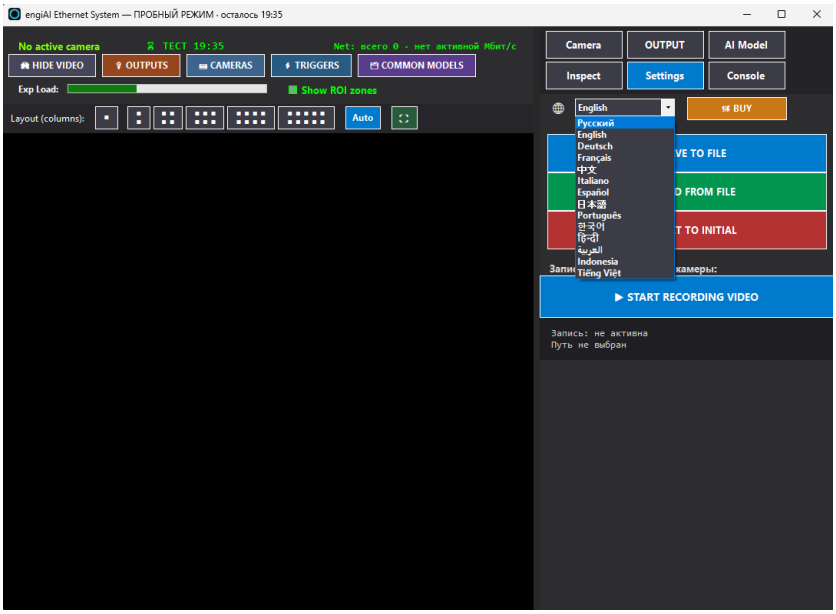
Tab	Purpose
Console	Event log. If something isn't working correctly, check this first.

Status bar

The status of the active camera is indicated above the buttons: `LIVE (SICK): 1920×1080 | 23,0 fps` – resolution and frame rate; `AI: GPU (CUDA)` – what the model considers to be optimal; `Net` – network load. The **Exp Load** bar shows what proportion of the frame interval is occupied by the shutter speed: if it reaches the rightmost position, the camera will not be able to maintain the specified frame rate.

Interface language

The language is selected in the "**Settings**" tab, located at the top of the window (a dropdown menu next to the globe icon). There are 14 languages available; the main window label changes immediately, while the labels of other windows change upon the next opening.

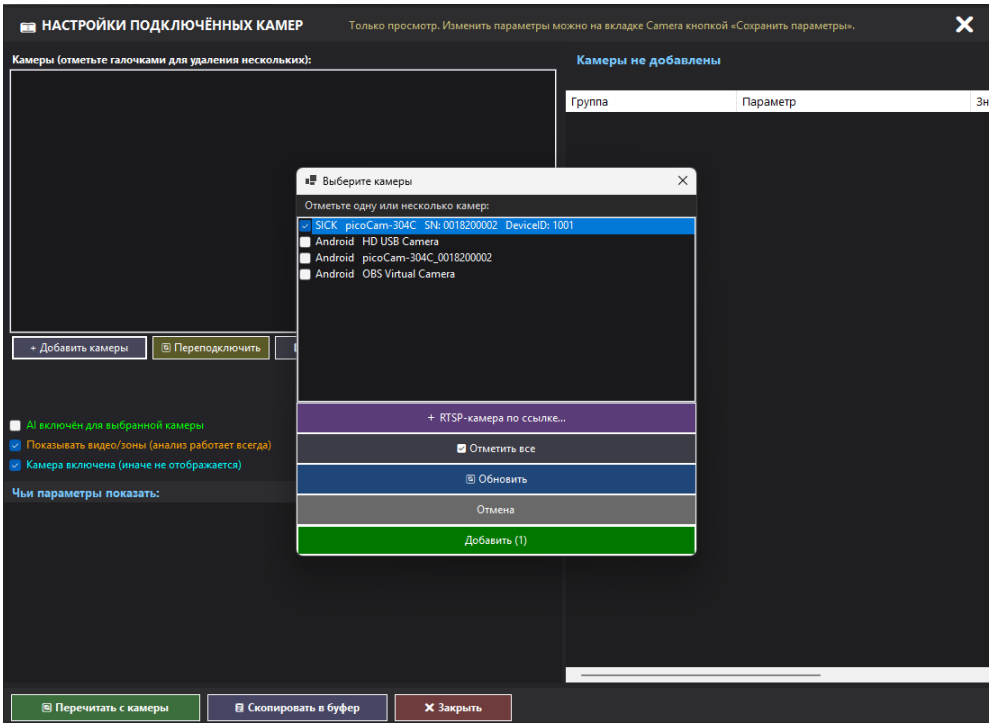


Settings tab: a dropdown list of languages.

Step 1: Connecting and Configuring the Camera

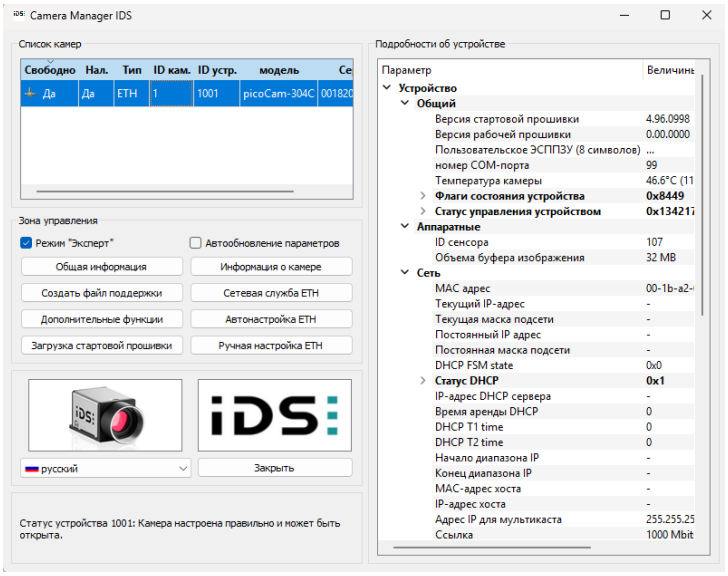
1.1. Add the Camera

1. Click on "**CAMERAS**" above the video. This will open the "Settings for Connected Cameras" window.
2. Click **+ Add Cameras**. The program will scan both the network and USB and display all found devices: SICK picoCam, HIKROBOT cameras compatible with GigE, USB, and RTSP.
3. Select the desired cameras by checking the boxes and then click "**Add**." While the cameras are connecting, a progress indicator will be displayed. Loading window – the program did not freeze.
4. The camera will appear in the list with its status indicated as either **LIVE** or **OFFLINE**, and its video will be displayed on the left side. of the main window.



CAMERA LIST → "Add Cameras": Devices found are marked with checkmarks.

> **Important.** The SICK picoCam camera should be visible in **IDS Camera Manager** (which comes with the uEye driver) with the status "properly configured". If it is not visible there, check the power supply, network cable, and IP address; the engiAI software will also not be able to see it.



IDS Camera Manager: SICK camera found, status is normal.

Checkboxes next to a list of cameras

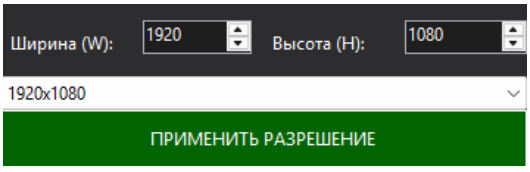
- **AI is enabled for the selected camera** – take a photo so that the camera records video, but not were analyzed.
- **Displaying videos/zones** – only affects the display; the analysis function always works.
- **Camera turned off** – completely disables the camera, causing it to disappear from the screen.

If the camera doesn't connect – click **Reconnect**. The HIKROBOT camera using the GigE protocol maintains a dedicated connection: after the program is unexpectedly closed, it remains in use for a short period, so please wait and reconnect. The **Network** line indicates whether there is network traffic.

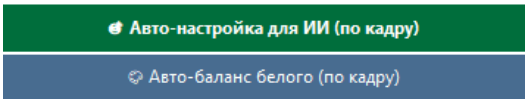
1.2. Configure the Image (Camera Tab)

This tab works with the camera, which is highlighted by a blue border. To switch to a different camera, simply click on its tile on the left.

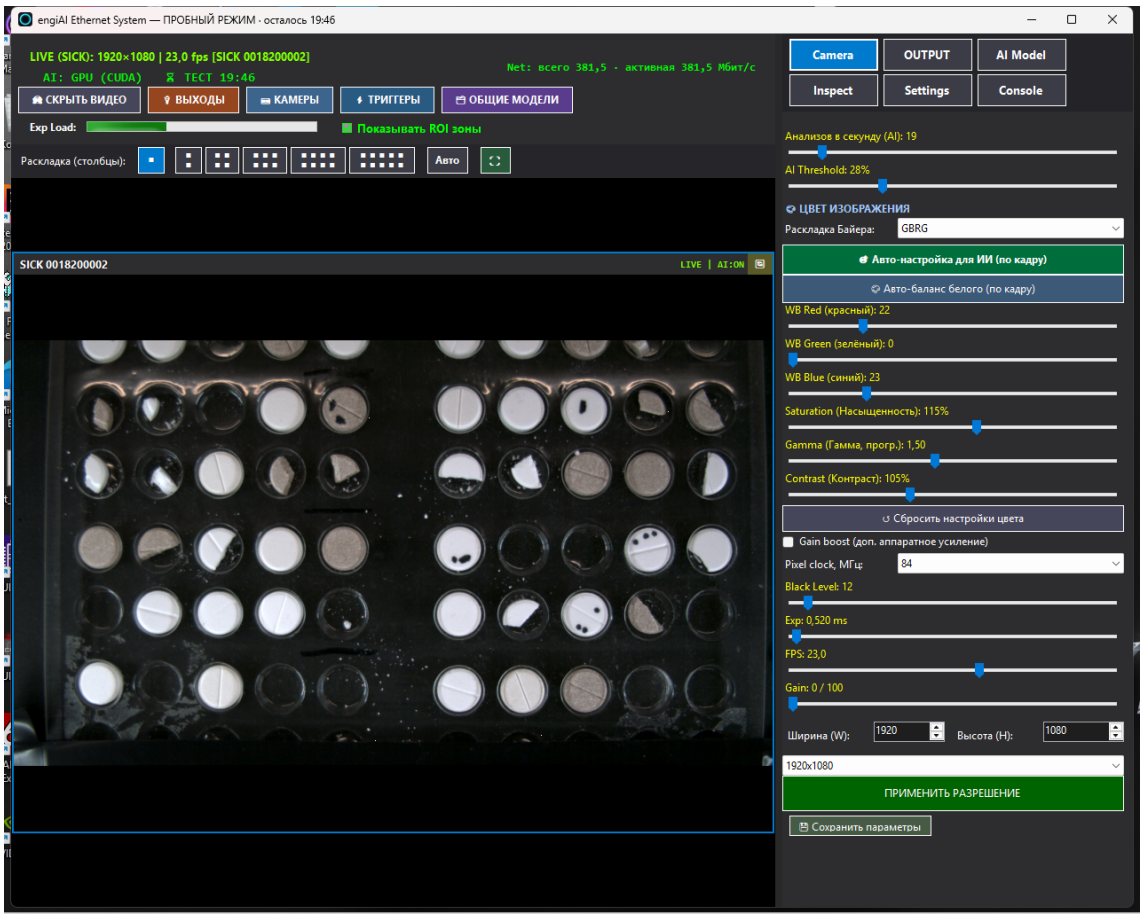
1. At the bottom of the tabs, enter the **width** and **height** of the frame (or select a preset from the list) and click. **APPLY THE RESOLUTION**. The higher the resolution, the more details the model will be able to see.
2. Click **Auto-Settings for AI (per frame)**. This button automatically adjusts the shutter speed, amplification, and balance. white and gray, ensuring that the product is clearly visible without glare. If necessary, provide separately. - **Auto white balance**.
3. If necessary, adjust the **Exposure** and **Gain** sliders manually. For moving subjects, The "dwell time" is the key parameter: the shorter the time, the less lubricant is needed.
4. Click **Save Settings**: the values will be saved within the camera and will be restored after it is turned off. nutrition.



Frame resolution and "APPLY SETTING" button.



Automatic configuration for AI and automatic white balance – one for each to the button.



The "Camera" tab: threshold, color, exposure, amplification, resolution.

What the parameters mean:

Parameter	What it does	What it affects
Analysis per second (AI)	Upper inspection speed limit	Lower speed = lower GPU load
AI Threshold	Model confidence threshold	Below the threshold, the result is not accepted
Bayer pattern, WB, Saturation, Gamma, Contrast	Image color (for color SICK)	Correct colors for tablets, packaging, and labels
Exp	Exposure time, ms	Shorter = less blurring of moving objects
Gain / Gain boost	Matrix signal amplification:	Increases brightness, but also adds noise.
Black Level	black level	lower brightness threshold
FPS	Frame rate in free mode	Not used in trigger mode
Width / Height	frame size	Smaller frame = higher frequency and less traffic

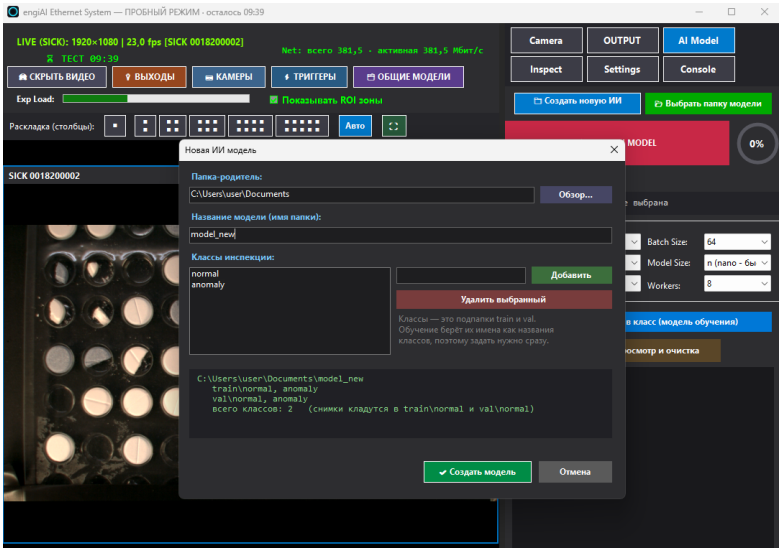
Important. When taking pictures for training, do so under the same conditions that the line will operate in: the same lighting, the same speed, and the same camera position. A model trained under different conditions will not work correctly on the line.

Step 2: New AI Model and its Classes

The model learns to distinguish between "good" and "defective" by examining small images – cropped frames. These crops are defined by **zones**, and the number of crops increases with **synthetic data** (through rotations and changes in lighting). The **class** is simply a folder within `train` and `val`: the number of folders determines how many answers the model can provide. Typically, there are two classes: `normal` (good) and `anomaly` (defective). The names are in Latin characters.

2.1. Create a Model

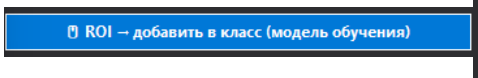
1. Click on the "**AI Model**" tab and click "**Create New AI Model**".
2. Specify the folder and model name (e.g., `model_new`). The program will create folders `train` and `val` within it.
3. The list of classes already includes `normal` and `anomaly`; you can add or remove classes directly from this list.
4. Click **Create Model**. If a model already exists, instead click **Select Model Folder**.



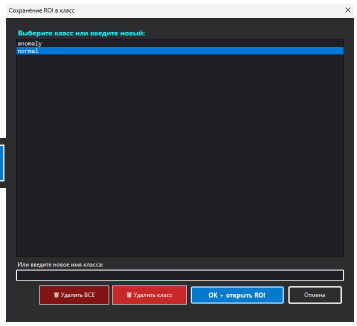
"New AI Model" dialog: folder, name, class list.

2.2. Open the Class Zone Editor

1. Click **ROI → Add to Class (Training Model)**.
2. In the list, select a class (starting with `anomaly`) or enter a new name at the bottom.
3. Click **OK + Open ROI** – this will open the ROI (Region of Interest) editor.



Button on the AI Model tab.



Class selection dialog: list, field for a new name, "OK + open ROI".

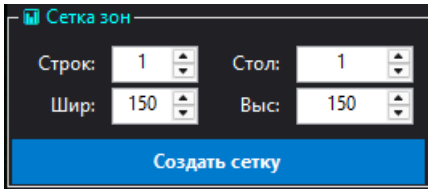
Confirmation of class selection.

Step 3: Zones, Images, and Synthetic Data

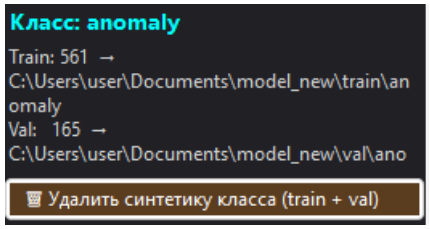
In the editor, you specify rectangles that will be used as training images. On the right, there is a panel for synthetic data, a grid of zones, a photo counter, and capture buttons. The window can be resized and moved; its size is saved.

3.1. Place Zones on the Product

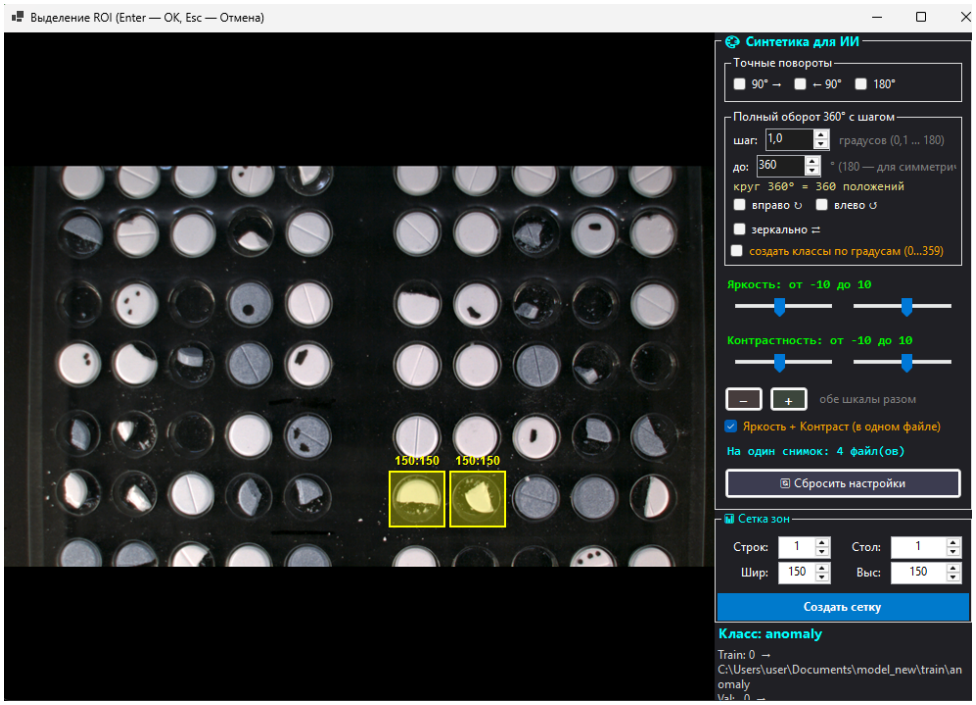
1. In the `**Grid of Zones**` group, specify the number of rows and columns, and the cell size. The default cell size is `**150x150**`. pixels – please specify this size as the "Image Size" when training.
2. Click **Create Grid**. A grid of yellow zones will appear; drag it to the desired location on the product.
3. For defective items, mark the defective tablets (broken, chipped) with the "defective" label, and for the class normal — in whole numbers.
4. **Color** sets the color of the border, **Remove** removes the area, **Remove ALL areas** clears the marking before transitioning to a different grade.



Grid structure: rows, columns, cell size 150x150.



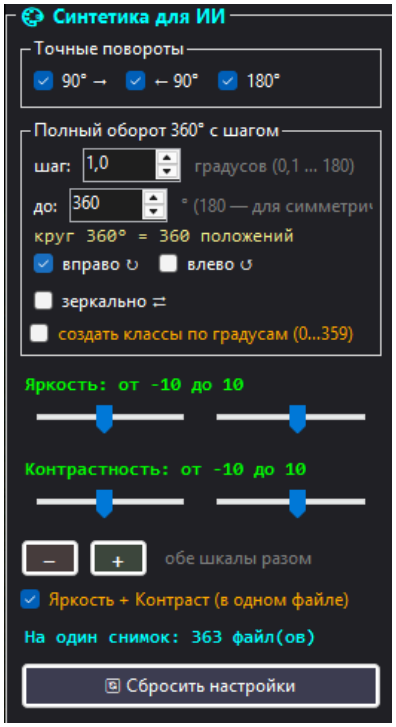
Class counter: How many images have already been added to the train and validation sets.



Editor for zones: two zones have been placed on broken white tablets (anomaly class).

3.2. Configure Synthetic Data

synthetic data – additional images that the program generates from each captured frame, so that the model can recognize the product as being rotated or differently lit. The **synthetic data** panel is located in the top right column of the editor. The line **For one image: N files** immediately shows how many will be generated.



Synthetic data panel: precise rotations, rotation 360° field of view, adjustable brightness and contrast ranges file counter

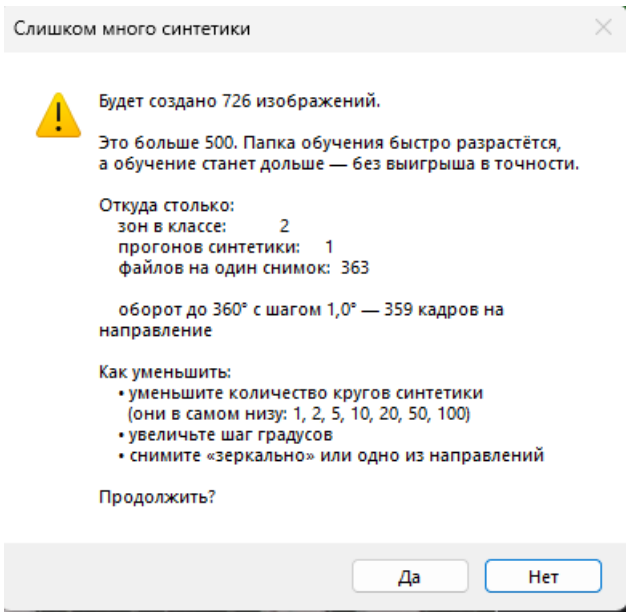
The number of synthetic data circles (1...50) and the "Take Snapshot" button.

Line conditions:	Brightness	Contrast	Rotation
Stable lighting, object remains in the same position	-5 to 5	-5 to 5	All objects are captured
"Erratic" lighting (window, lamps of different ages)	-20 to 20	-10 to 10	All objects are captured
Object appears in any position	-10 to 10	-10 to 10	360° rotation with a step
The part is symmetrical, but it can lie on its side.	-10 to 10	-10 to 10	only 180 degrees

Tip. Don't focus on quantity. Five hundred variations of the same image won't be as valuable as fifty **different** examples. Synthetic data can improve robustness to lighting and rotation, but it doesn't introduce new types of defects. Generally, 1000 images are sufficient for each class.

3.3. Take Images

1. Select the number of synthetic data circles (**1**) at the bottom and click **Take Snapshot**.
2. If the settings result in more than 500 files, a warning message will appear, displaying the exact number. according to the recommendation, reduce it. This is normal – click **Yes**.
3. The "**STOP**" button interrupts the generation process at any point; the files that have been created remain.
4. Check the **Class** counter: it shows how many files are currently in the train and val sets. The **Delete** button **synthetic data of class** only removes files created by the program (`syn_...`), not the actual frames (`real_...`). touches.



Warning about the large volume of synthetic data: where the number came from and how to reduce it.

Where do the photos go: Train and Val

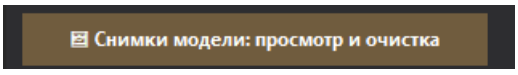
A validation set (`val`) is needed to fairly evaluate the trained model. By default, each image goes into the training set (`train`), and with a 20% probability, a copy of it also goes into the validation set (`val`) (the checkbox **Include in Val** has a **20% probability**). The option **Include every 4th in Val** does the same thing, but in a more regular manner; only one of these methods can be selected.

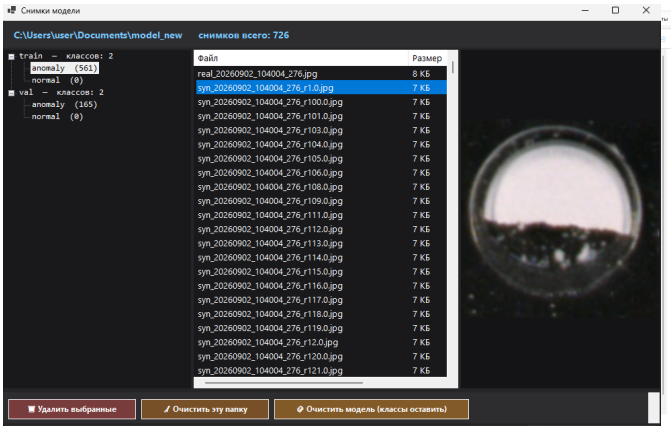
Button / Checkbox	What it does
Take a picture	One frame per click, using all synthetic data
Create screenshots	a series of consecutive shots; a source of timing is needed – either "per frame/second" or "Encoder"
Shot/frame	The series is recorded at a specified frame rate; it is disabled by default.
Regarding the encoder:	Images taken every N steps of the encoder – the images are evenly spaced along the product's path.
Save in Train	Images are added to the training dataset.
In Val, there is a 20% chance / a 1 in 4 chance.	How to select copies for the validation dataset.
1 / 2 / 5 / 10 / 20 / 50	How many synthetic data sets to create for each image.

> **Settings.** All the settings in this window—grid, synthetic data, shooting mode, and the Train/Val split—are saved **for the model.** When moving to the next class of the same model, no need to reconfigure anything.

3.4. Check what has been entered

The "**Model Images: View and Clean**" button on the AI Model tab opens a directory tree `train` / `val`, showing the number of files in each class. Here, you can view the images, delete unwanted ones, clear a specific class, or clear the entire model, leaving the folders empty.



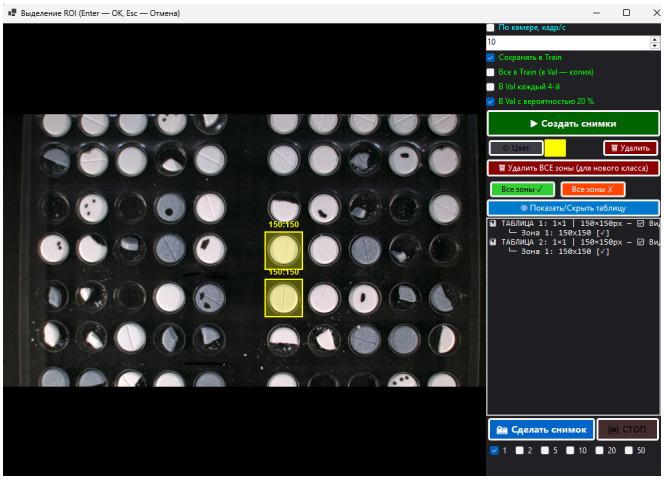


"Model Images" window: Anomaly class, 561 files in the training set.

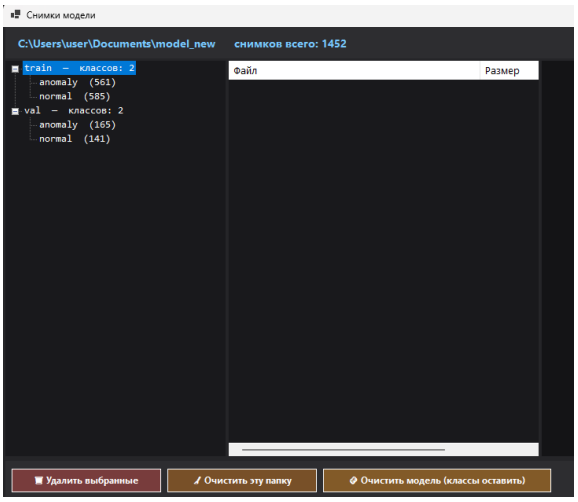
The same folder in the Windows file explorer: 150x150 pixel screenshots with turns.

3.5. Repeat for the "normal" class

Click again on **ROI** → **Add to class (training model)**, select `normal`, then click **OK + Open**.
ROI, place the samples on whole tablets and take pictures. There may be more than two classes —
They are added using the same dialogue and by inserting a new name.



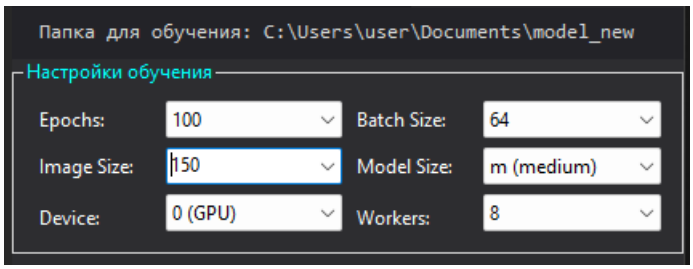
Excellent: the effects are noticeable even on a single tablet.



Summary of the dataset: anomalies – 561, normal – 585 images in the training set.

Step 4. Model Training

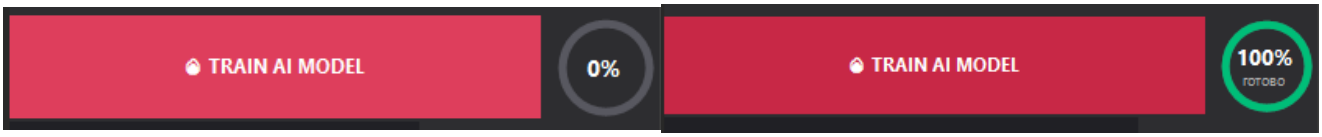
Once the classes are filled, return to the "**AI Model**" tab. The "**Training Settings**" group defines how the model will be trained.



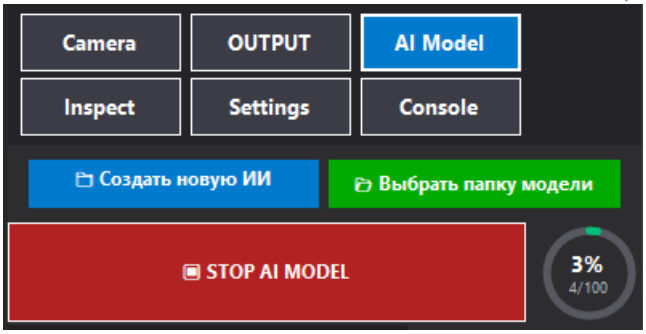
Training settings. Image Size = 150 – this represents the size of the grid cell.

Parameter	What it means	How to choose
Epochs	How many times will the model process the entire dataset?	100 for testing, 300 for the working model
Batch Size	How many images are processed per step?	64 is usually sufficient; if there is a lack of video memory, reduce it to
Image Size	What size are the crops reduced to?	The size of the grid cell (150), preferably a square; use this to inspect the network
Model Size	Network size: n (nano), s (small), m (medium)	n – faster, m – more accurate
Device	On what to train	0 (GPU), if there is an NVIDIA graphics card, otherwise CPU
Workers	Number of data preparation threads	8 for a standard computer

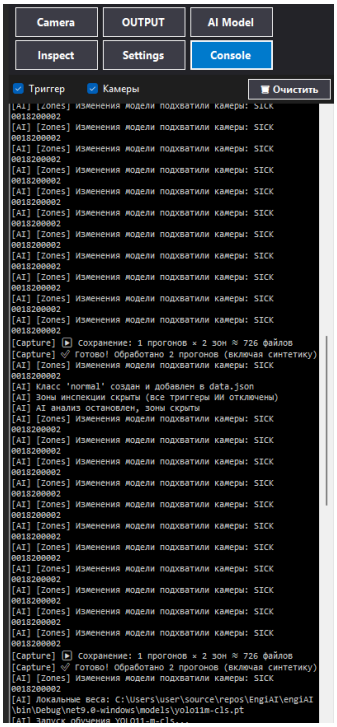
1. Click **TRAIN AI MODEL**. The button will turn red **STOP AI MODEL**, allowing you to stop the training.
2. The ring to the right of the button indicates the percentage of completion based on the number of epochs completed.
3. The training process is recorded in the "**Console**" tab.
4. Upon completion, a message "Training and export complete" will appear, and in the model folder, a file will be created. `best.onnx`. This is the trained model.



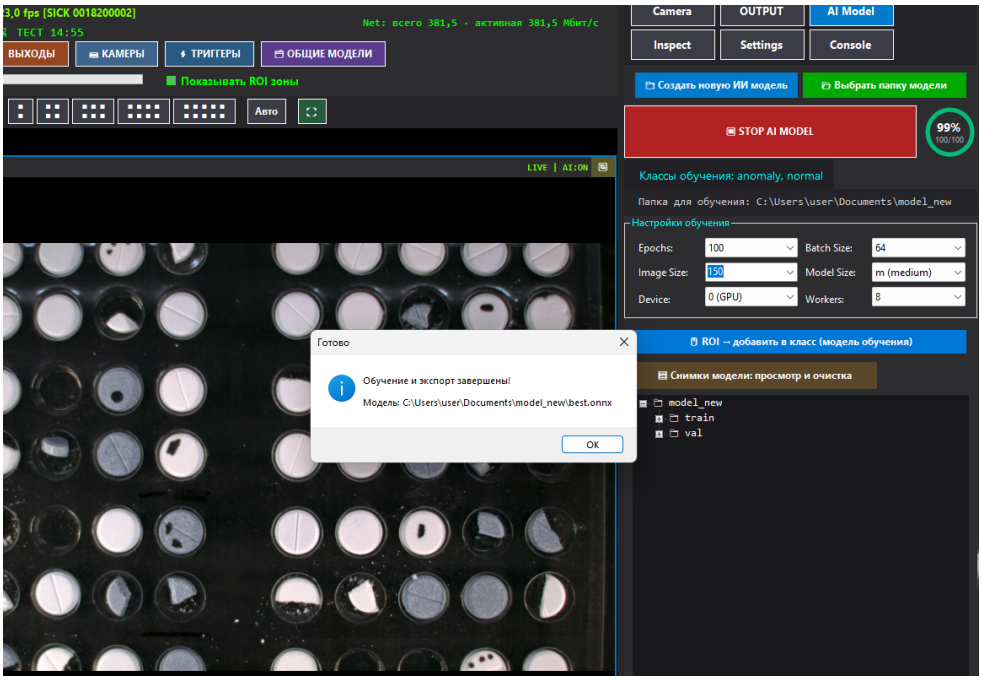
Start: TRAIN AI MODEL, 0 %. Complete: 100 %.



Training is in progress: the button became STOP AI MODEL, the ring shows 3 % (4 out of 100 epochs).



Console: training lines.



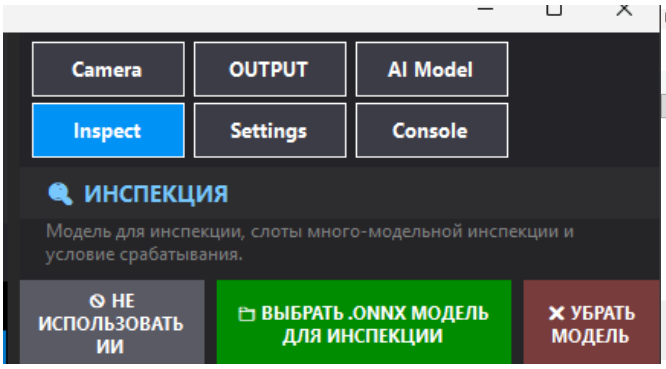
Message indicating the completion of model training and export.

Step 5. Inspection

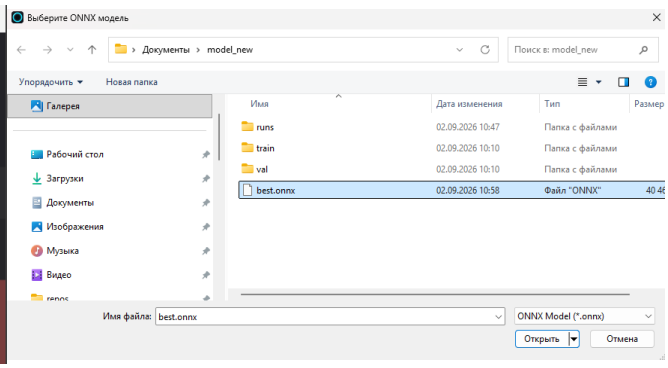
All the work is done on the "**Inspect**" tab. First, assign the model, then mark the inspection areas and choose which signal to use to capture frames.

5.1. Assign the model

1. Click on the "**Inspect**" tab and click "**SELECT .ONNX MODEL FOR INSPECTION**".
2. Specify the file `best.onnx` in the model directory.
3. The "**Inspection Classes**" line indicates that the model can distinguish between different categories (e.g., 1 = anomaly, 2 = normal).



Inspect tab: three buttons for controlling the model.

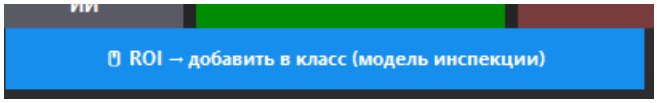


Select the file best.onnx in the model folder.

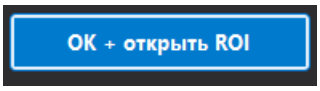
DO NOT USE AI temporarily disables analysis without deleting settings; **REMOVE MODEL** removes
The model is completely finished.

5.2. Define inspection zones

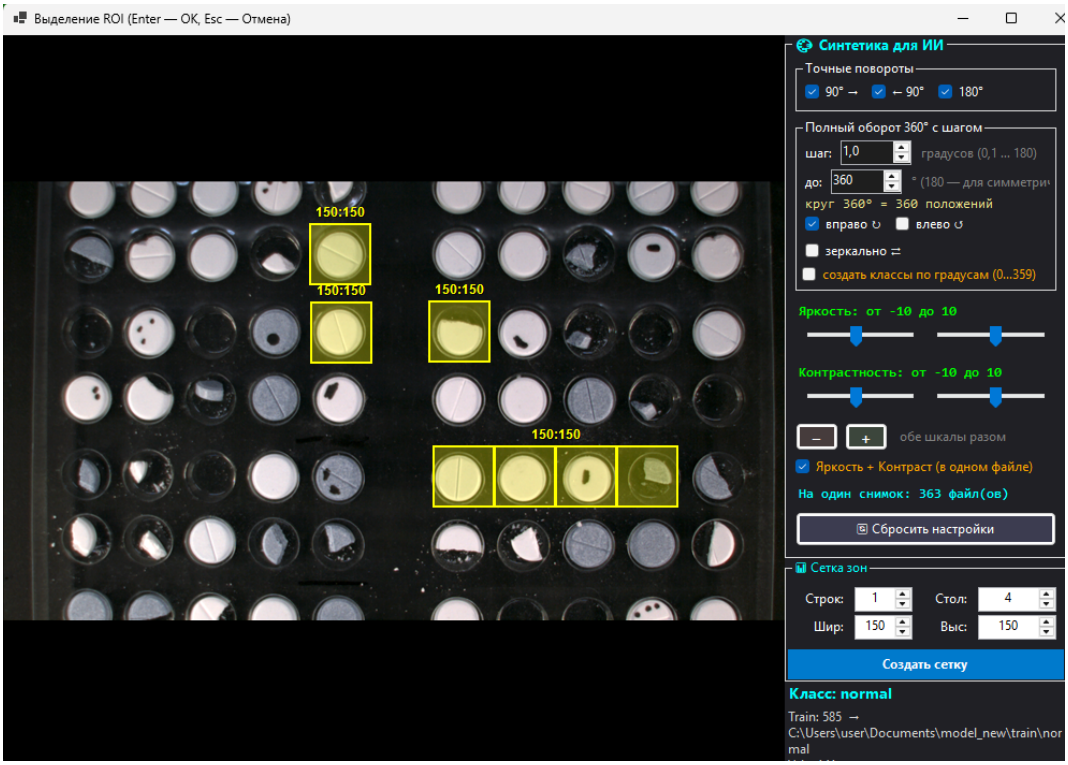
1. Click **ROI → Add to Class (Inspection Model)**.
2. In the dialogue, select **any** class – it is irrelevant for inspection, as all areas are common.
3. Click "**OK + Open ROI**" and define the areas where the products will be located. The inspection areas should be... Each camera has its own specific characteristics.



Inspection zone marking button.



Confirmation.



Inspection areas: The nets were placed around several tablets.

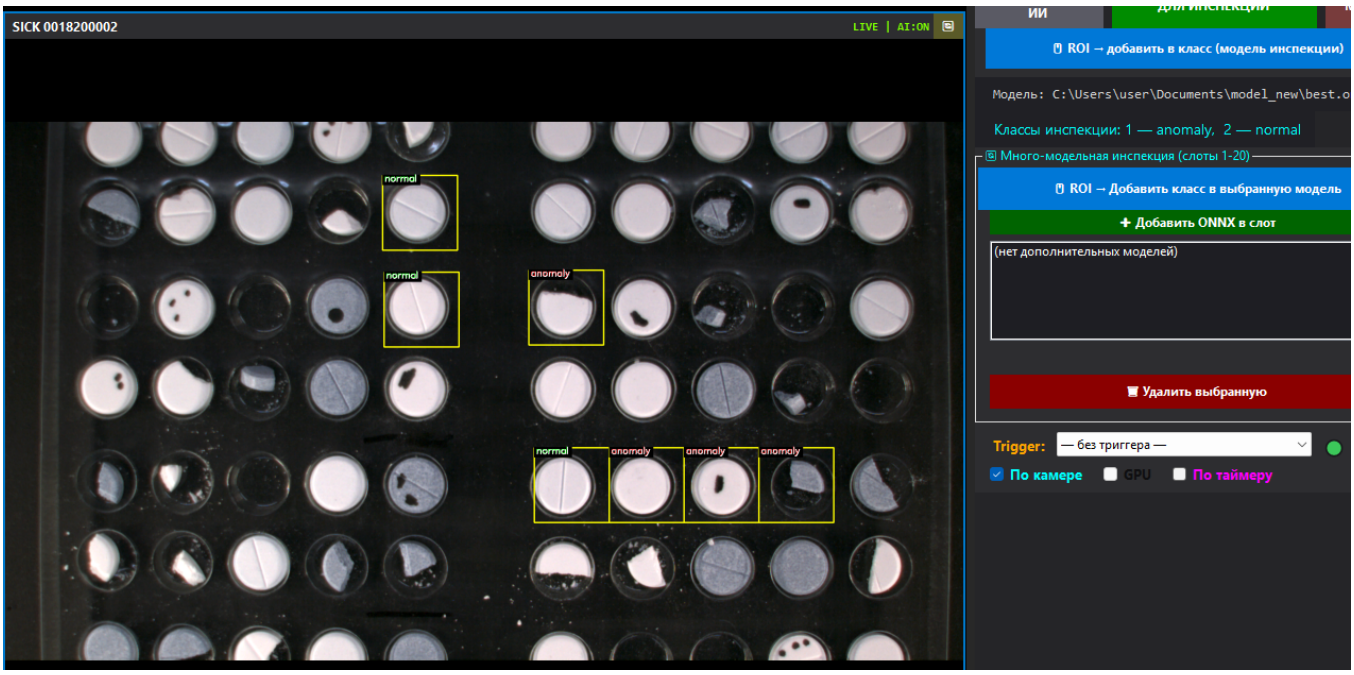
5.3. Choose the signal to analyze

The source of the speed always comes from one place: the program removes the previous source, regardless of whether it's a new one. For the initial check, select the **Camera** option – the analysis will be performed on each frame.

You have an NVIDIA graphics card: make sure the GPU is enabled.

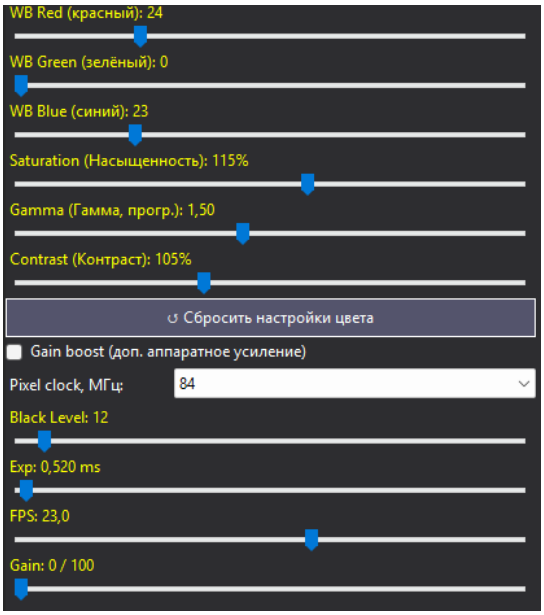
Analysis on the processor: GPU removed.

Method	How to enable	When to use
By camera	"By camera" checkbox	continuous monitoring; pace controlled by the "Analysis" slider "in seconds" on the Camera tab
By timer	checkbox "Set Timer" and interval, ms	Precise and infrequent control – for example, once per second
Regarding the encoder:	"Triggered by Encoder" checkbox	Conveyor with encoder: analysis linked to the traveled path
By trigger	Select a trigger in the list	External sensor on the camera input: frame is captured at the moment the product arrives – the most accurate method



Inspection is working: each zone is labeled with a class – normal with a hole, and the others are anomaly.

Tip. If the result is uncertain, return to the **Camera** tab and adjust the exposure, amplification, and color settings – the image should look the same when inspected as it did when the photo was taken.

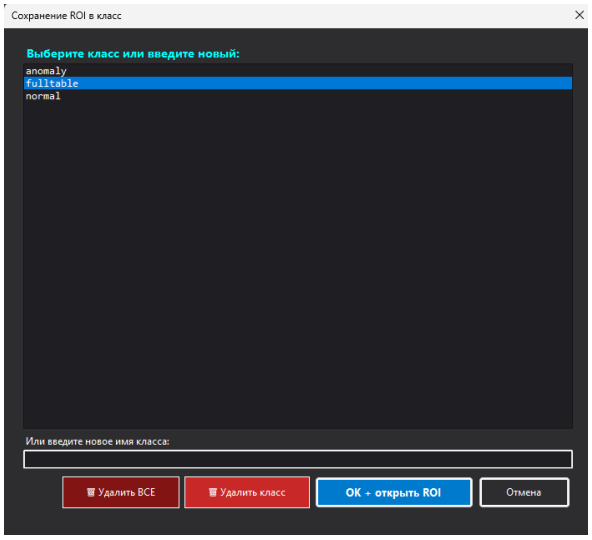


Color and exposure parameters affect the accuracy of the analysis.

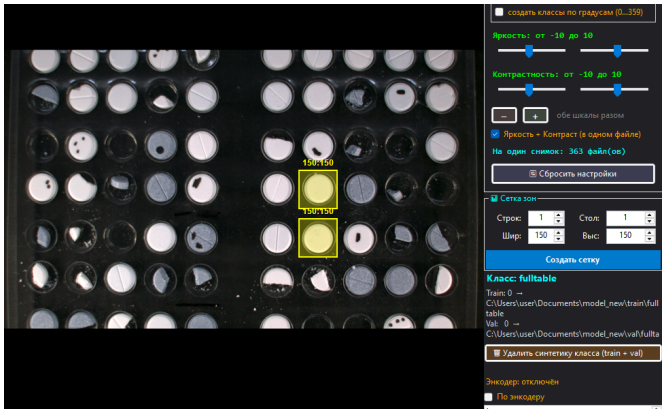
Add the class and retrain the model

The model can be extended: for example, in addition to `normal` and `anomaly`, you can add a class for a whole tablet `fulltable`. The order is the same as in the initial setup.

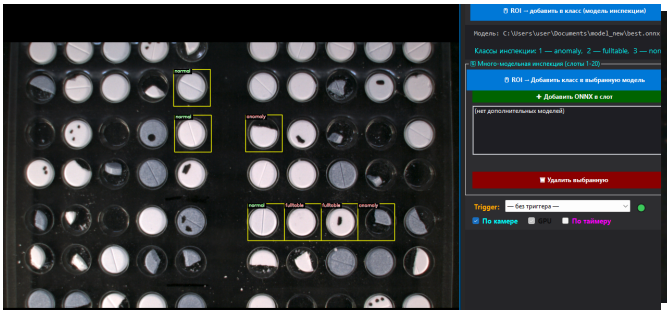
1. On the "AI Model" tab, click "**ROI → Add to Class (Training Model)**", and enter a new name. In the lower part of the dialogue, click "**OK + Open ROI**".
2. Mark the zones on the new class of products and take pictures.
3. Click **TRAIN AI MODEL** and wait for 100%. The file `best.onnx` will be overwritten.
4. On the "Inspect" tab, re-select `best.onnx" – a third class will appear in the class list.



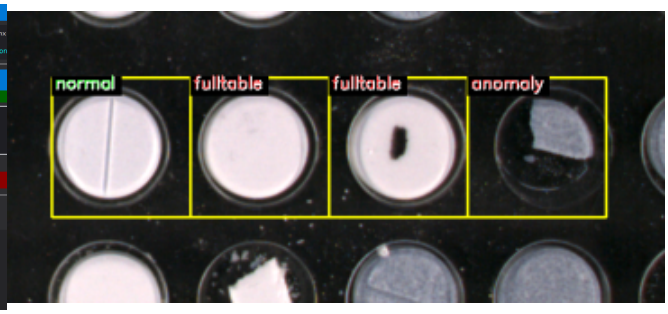
New class "fulltable" in the list.



Areas on whole tablets for the fulltable class.



After retraining, the model can distinguish between three classes.



Large: normal, fulltable, fulltable, anomaly.

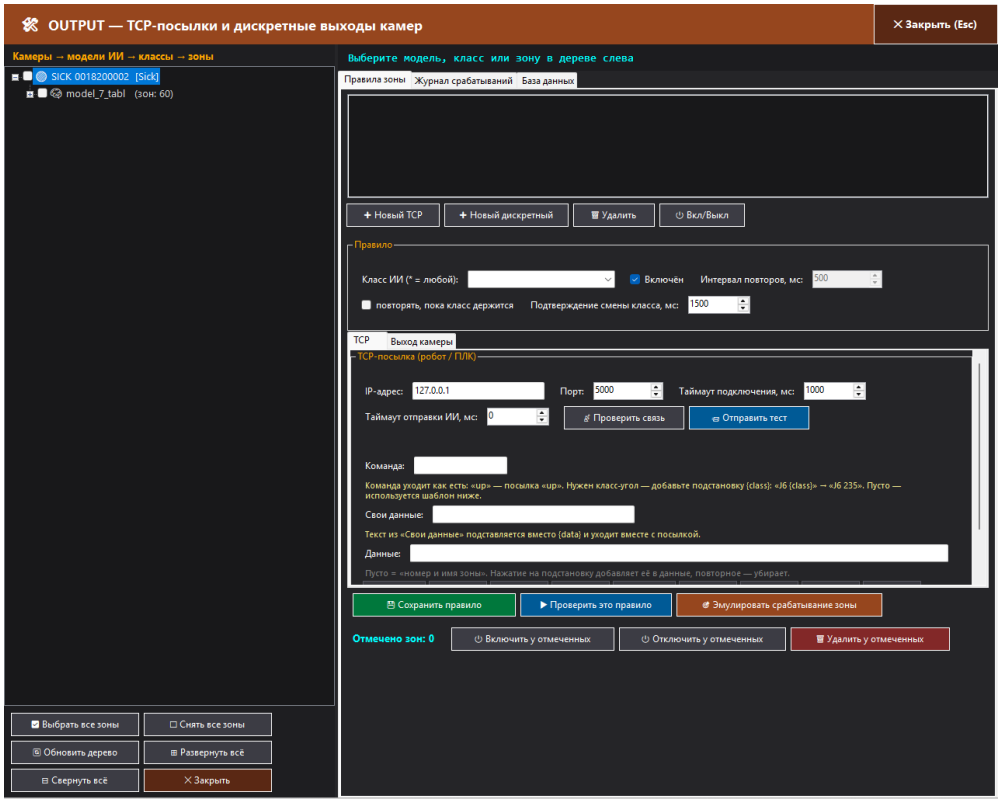
This allows for the creation of numerous classes and inspection zones. The program supports:
Multi-modality (multiple models on a single camera, to prevent similar classes from blending together – group) "Multi-model inspection" (under the "Inspect" tab), encompassing **multi-camera** and **multi-zone** capabilities.

What's next: rejector, triggers, general models

7.1. rejector (OUTPUT tab)

The rejector converts the inspection result into an action: either sending a network message or switching the camera's discrete output. The left side shows the tree structure: `camera → model → class → zone; a rule is created for a specific zone of a specific camera.

1. Mark the area on the tree and click "**+ New TCP**" or "**+ New Discrete**".
2. In the `**Rule**` group, select the `**AI Class**` that should trigger (e.g., anomaly), and the interval. repetitions.
3. For TCP communication, specify the robot or PLC's IP address and port, the command, and the data. Placeholders like `{zoneId}` are available for use. {zone} {class} {conf} {camera} {model} {time} {date} and {data} (field "Your Data") – clicking on It adds the substitution to the data.
4. To achieve a discrete output, select the line, level, and pulse duration.
5. **Verify connection** / **Send test** / **Simulate zone activation** – testing the entire chain without expecting a defect.
6. Click "**Save rule**" – this is necessary for the rule to function. The actions triggered by the rule can be viewed on the "Actions" tab. "Log of operations"



Output window: a tree-like structure on the left, with a TCP-sending rule on the right.

7.2. Trigger from an external sensor (TRIGGER window)

Main mode for the conveyor: the camera captures a frame based on the electrical signal from the sensor at its input – the most precise moment of capture. In the **TRIGGER** window, select the sensor equipment, camera connector (HIKROBOT or SICK 6 pin), camera, and sensor output type (PNP – standard case, NPN – requires a pull-up resistor), then click **Connect** and assemble the circuit according to the diagram. Then, on the Inspect tab, select the trigger in the **Trigger** list and the front **analysis with + on input**.

Important. The "male" connectors on the camera and the "female" connectors on the cable are mirrored. The only reference point is the **key** (a large opening in the connector). A faulty connection means that the power and output are shorted together. The **OUTPUTS** button shows the pinout and how to test with a multimeter.

7.3. General models and Settings

- **GENERAL MODELS** – situations where a single model should be present on multiple cameras simultaneously: a set of It is retained under the specified name and assigned to the cameras.
- **Settings** → **SAVE TO FILE** exports all program settings (camera, model, rules) — Convenient for reinstalling or transferring to a second computer; **DOWNLOAD FROM FILE** returns them.
- This also includes recording video from an active camera and purchasing a license.
- **Console** – Journal: Camera connections, training progress, rejector activation, warnings. If there's a problem, please check this first.

A short checklist from scratch

If you don't have time to read everything, here is the complete sequence.

1. **CAMERAS** → "Add Cameras" → Select your cameras → Add.
2. Tab **Camera**: Set resolution → "Automatic AI settings" → "Save settings".
3. Tab **AI Model** → "Create a new AI model" → enter a name.
4. «ROI → Add to class (training model) → Class `anomaly` → Create grid 150x150 → Place»
   Areas on defective items → "Take a picture."
5. The same applies to the `normal` class for compliant products.
6. Training settings: Image Size 150 → **TRAIN AI MODEL** → wait for 100% → obtain `best.onnx`.
7. Tab **Inspect** → "SELECT .ONNX MODEL FOR INSPECTION" → specify `best.onnx`.
8. «ROI → Add to inspection class (inspection model) → Mark inspection zones.»
9. Set the "In-Camera" (or choose a trigger) and ensure that the correct settings are applied to the zones.
   classes.
10. **OUTPUT** → select zone → rule for anomaly class → "Simulate zone activation" →
    "Maintain the rule."

If something goes wrong:

Symptom	What to check
Camera does not appear in the list	power and cable; for SICK — IDS Camera Manager; for HIKROBOT after wait for the emergency shutdown and "reconnect"
The image is dark/gray/has incorrect colors.	Camera → "Automatic settings for AI"; for SICK color cameras – Bayer matrix and white balance
The model is confusing the classes.	There are few different items; the photos were taken under different lighting; Image Size is not equal to the cell size
Learning is declining or progressing very slowly.	Device = CPU if no NVIDIA graphics card is present; reduce Batch Size; check the Console
The inspection system is not functioning correctly.	Was best.onnx selected? Are there inspection zones? Is the speed source (Po) enabled? (camera / timer / trigger); The AI threshold is not too high
The "rejector" rule is silent.	"Save rule" checkbox: is it checked? Rule class matches: the model's response