ITSCAM 600
Integration Manual

PUMATRONIX

Pumatronix Equipamentos Eletrônicos Ltda.

Rua Bartolomeu Lourenço de Gusmão, 1970. Curitiba, Brasil

Copyright 2020 Pumatronix Equipamentos Eletrônicos Ltda.

Visit our website http://www.pumatronix.com

Please send feedback on this document to suporte@pumatronix.com

Information contained in this document is subject to change without notice.

Pumatronix reserves the right to modify or improve this material without obligation to notify you of changes or improvements.

Change History

Date	Revision	Content updated
28/06/2022	1.0	Initial Version

Overview

This document aims to guide the developer in the use of the operating interfaces available for the ITSCAM 600, which enable configuring the device behavior. In this document are detailed the options available through the web interface, the Pumatronix Protocol, or the Cougar Protocol.

⚠️ **According to the firmware version applied to the device accessed, the web access interface is differentiated, and some functions can be made available only in the most current versions**

Figure 1 - ITSCAM 600 Line with CS Mount Lenses

Figure 2 - ITSCAM 600 Line with motorized lenses

Summary

1. Web Interface

The ITSCAM 600 has a web interface for evaluating the images generated and adjust settings. Access to the interface requires that they be informed:

Usuário	admin
Senha	1234
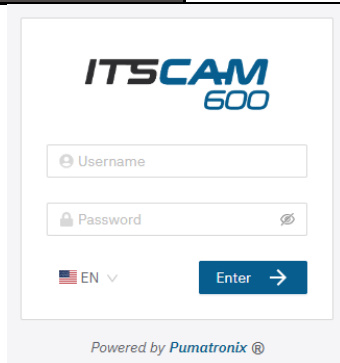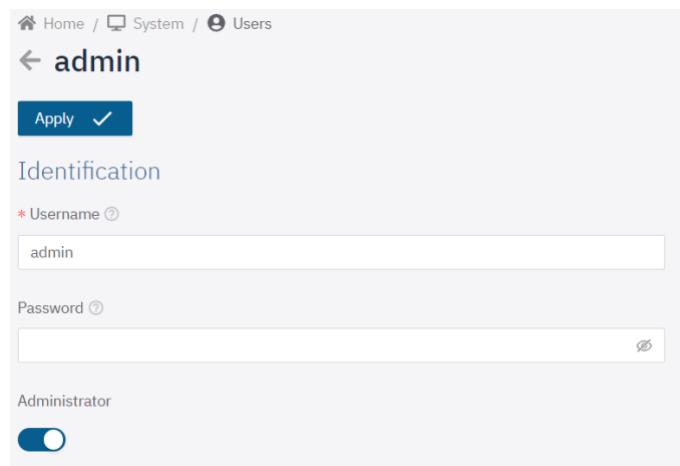
As a form of security, it is recommended to change the default device password after first access. The use of a *Configuration Assistant Equipment* is required to perform the password exchange.

1) Enter the factory default IP address of ITSCAM 600 in the browser;
2) Enter *admin* Username and Password *1234*;
3) Access the *System > Users* menu;
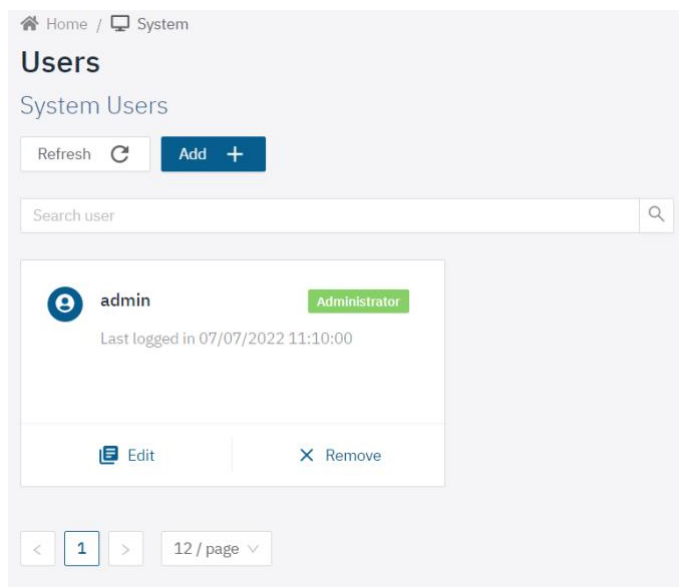
4) Click on *Edit*;
5) Enter a *Name* and a new password;

6) Click on the *Apply* button.

Help in the Web Interface

If there is doubt about any configuration in the web interface, there is the help symbol which, when positioning the cursor above, displays an explanation, examples, or the recommended configuration for the ITSCAM 600.
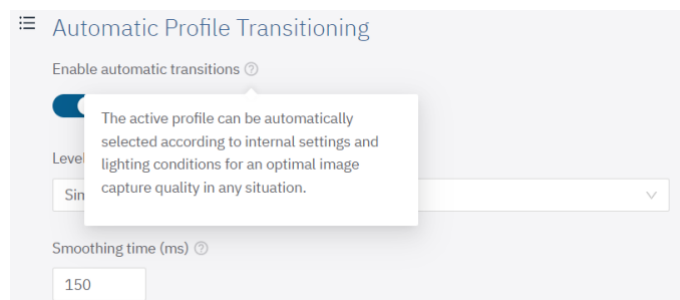
Figure 3 – Displaying help when positioning the cursor over it
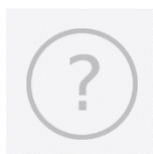
Figure 4 – Help Symbol

Tela Inicial Home Screen
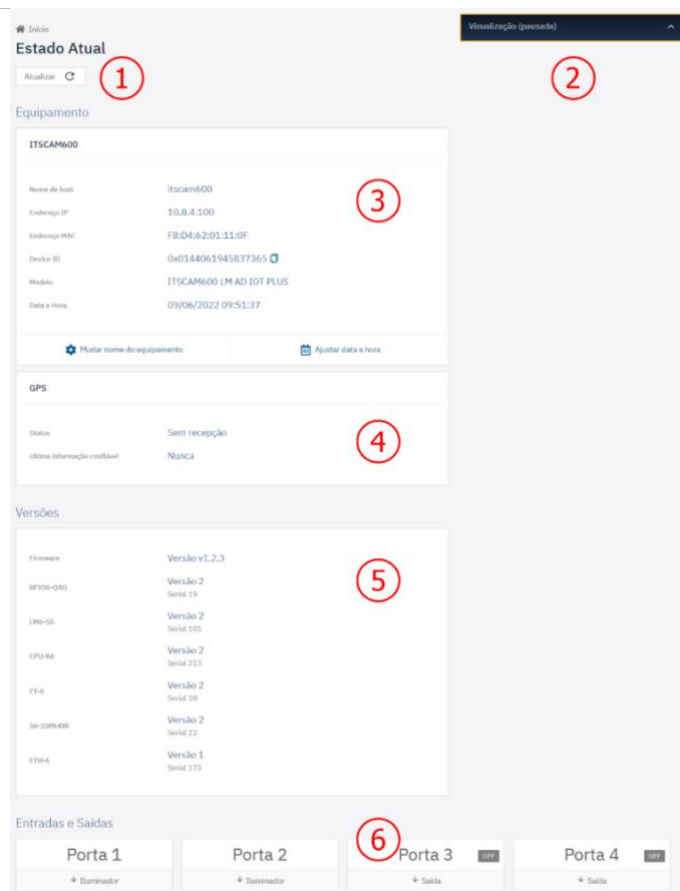
Figure 5 - 1) Refresh button; 2) Live Feed floating window; 3) Device information; 4) GPS location data; 5) Firmware and Hardware Version Information; 6) Status of connections on IO Ports.

GPS
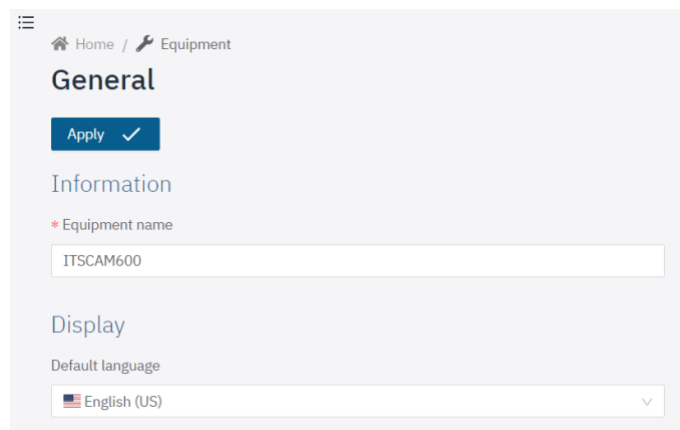
When the device can pin to the network, the main Geolocation information is displayed:

Figure 6 - GPS Geolocation Information

Changing the Interface Language

The ITSCAM 600 web interface can be displayed in Portuguese, English or Spanish. This setting is available in the *Equipment > General* menu.

Figure 7 - Equipment > General menu screen

⚠️ **The steps to configure the ITSCAM 600 device are presented in the ideal order of adjustments and must be carried out after physical installation, respecting the Prerequisites and the Necessary Conditions for installation presented in the Product Manual**

Network Setup

To perform the network configuration, it is necessary to use an *Auxiliary Setup Equipment*.

1) Enter the ITSCAM 600 factory default IP address in the browser;
2) Inform *admin* user and password *1234*;
3) Access the menu: *Equipment > Network*;
4) Access the respective tab to the network that will be configured.

The home screen of the network configuration allows you to view the Status of the networks and the diagram of the configured Routes:
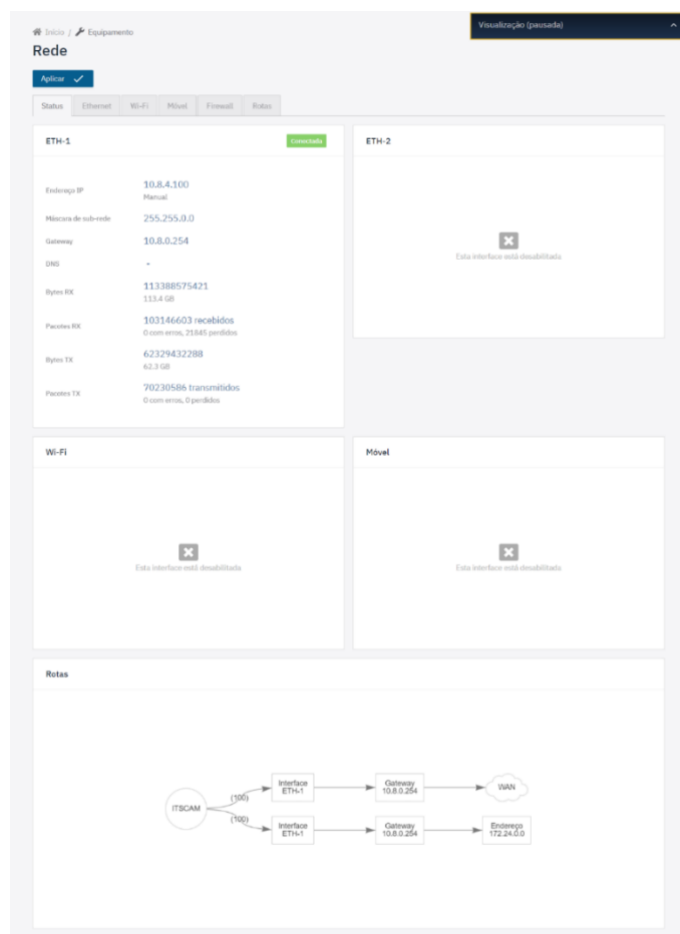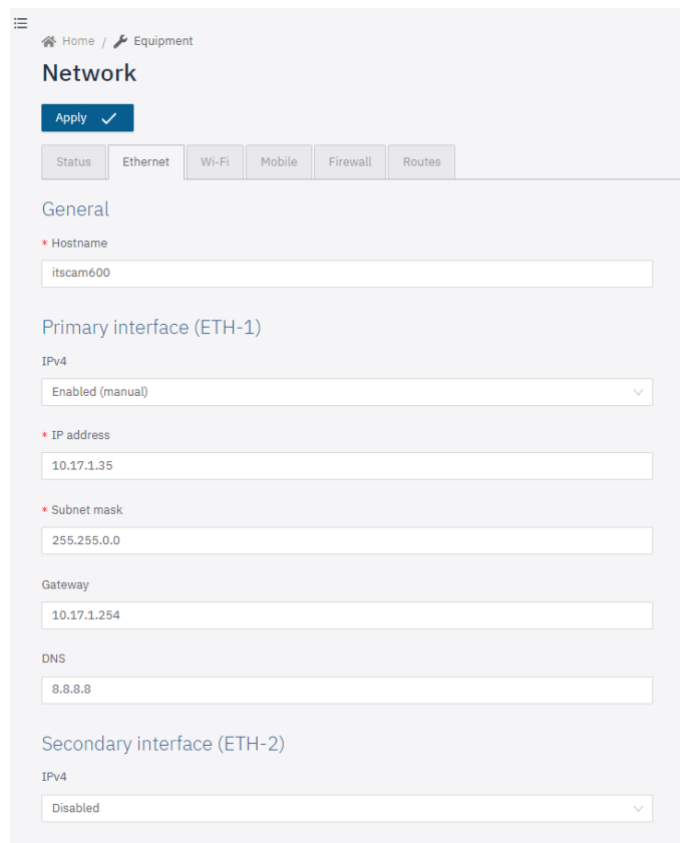
Figure 8 –Network configuration initial screen

Ethernet Connection

1) Clique na aba Ethernet; Click on the *Ethernet* tab;
2) Identify the network in *Hostname*;
3) Fill in the *Primary Interface (ETH-1)* data when the connection is made in *ETH-1*:

4) Click on the *Apply* button after confirming the entered data.

Wi-Fi Connection

1) Click on the *Wi-Fi tab;*
2) Enable Wi-Fi by clicking on *Enabled*;
3) Select the *Station (STA)* option in *Operation Mode* to connect to an existing Wi-Fi network;
4) Click on the *SSID* field and select the Wi-Fi network that will be used;
5) Select in *Authentication* the Authentication protocol to be used: *Open (without authentication)*, *WEP* or *WPA/WPA2 PSK*;
6) Enter the *Password* for access to the selected Wi-Fi network;
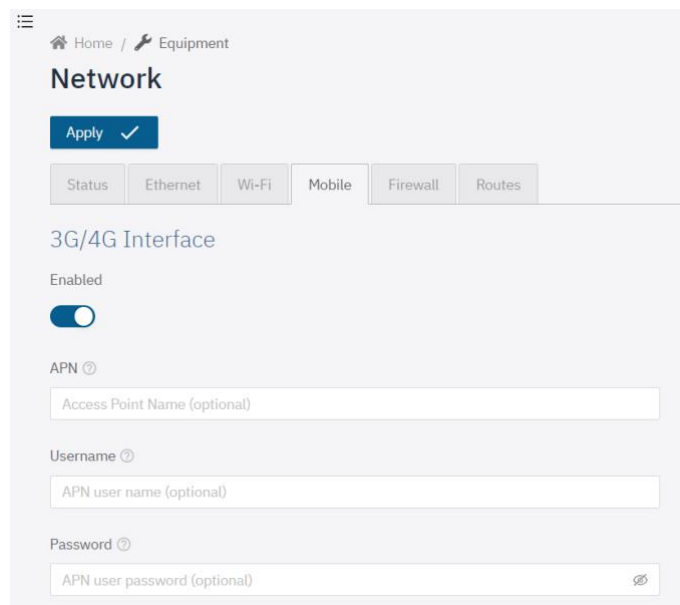
7) Click on the *Apply* button after confirming the entered data.

Select the *Access Point (AP)* option in the *Operation Mode* field only when the device is used as an access point to an available Wi-Fi network:

1) Enter the SSID identification data, *Country*, *Channel* and *Authentication* that will be applied in the Wi-Fi connection distribution;
2) Enter the *IP Address* and *Subnet Mask* of the *DHCP Server* by determining the address range for equipment that connects to the *Access point*.
3) Click on the *Apply* button after confirming the entered data.

3G/4G Connection
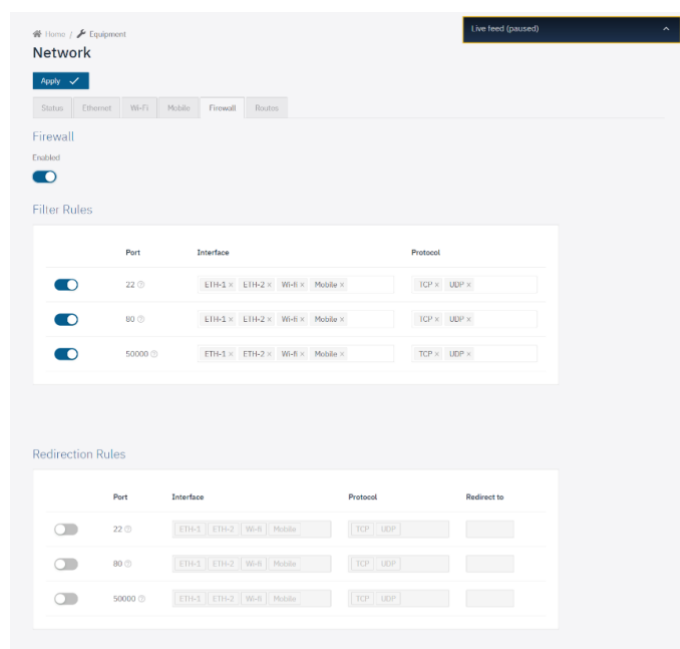
1) Click on the *Mobile tab;*
2) Enable 3G/4G by clicking on *Enabled*;
3) Fill in the data of the APN, if there is no automatic detection. It is recommended that Claro customers enter the APN manually:
   a. *APN*: claro.com.br
   b. *Username*: claro
   c. *Password*: claro

4) Click on the *Apply* button after confirming the entered data.

Firewall Settings

1) Click on the Firewall tab;
2) Enable Firewall by clicking on *Enabled*;
3) Fill in the *Filter Rules* and/or *Redirection Rules*;

4) Click on the *Apply* button after confirming the entered data.

Route Settings

Configure *Routes* when remote or not connected IP subnets need to be accessed directly to the network interface, which can be accessed using the default Gateway or the specified route displayed in the *Diagrams*.

1) Click on the *Routes* tab;

Revision 1.0
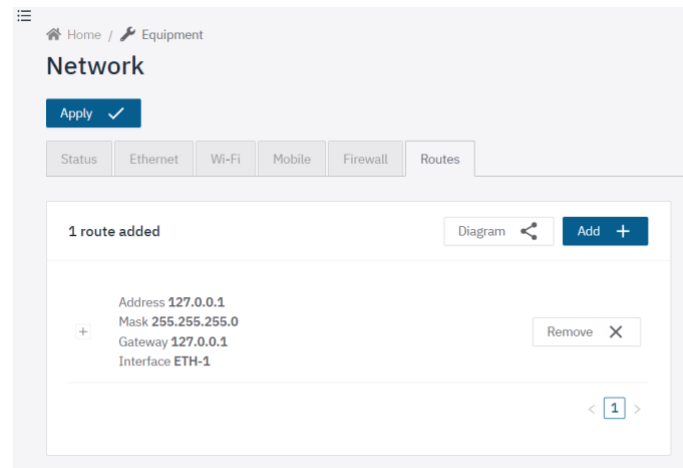
2) Click on the *Add +* button to add new routes;
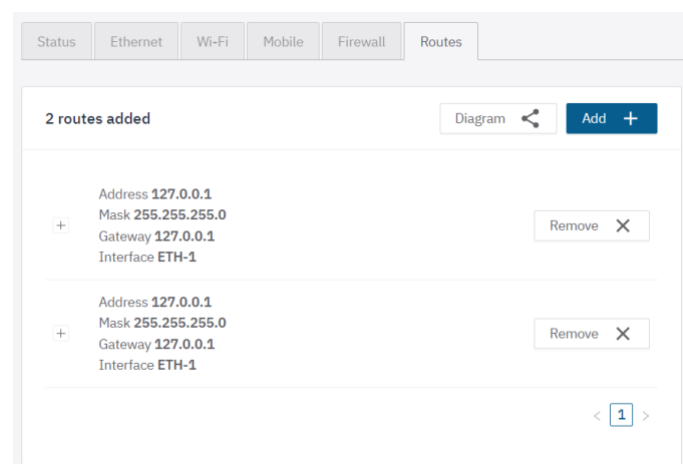
3) Click on "+" next to the *Address*, *Mask*, *Gateway*, and *Interface* information to edit it;
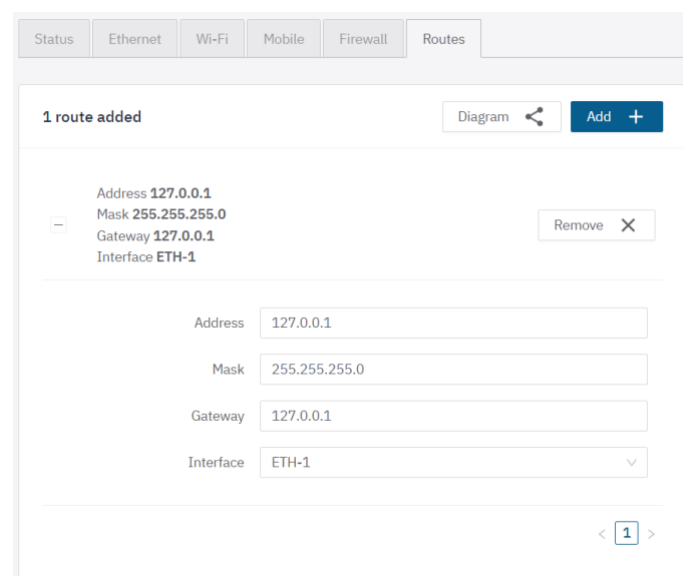
4) Click on the diagram button to view the *Route Diagram*;

Routes Diagram   X

5) Click on the *Apply* button after checking the information.

Access Management

The ITSCAM 600 allows greater access control and changes to the equipment, as multiple users can be created. Users can display Administrator or *Operator* access profile. In the case of users configured as *Administrator*, it is allowed to configure the equipment, users, and image viewing. Users with *Operator* profile can view images and settings.

Users have a unique identifier (containing between 4 and 200 characters, with letters and numbers, without space) and a password. For the management of active users, go to the menu *System > Users*.
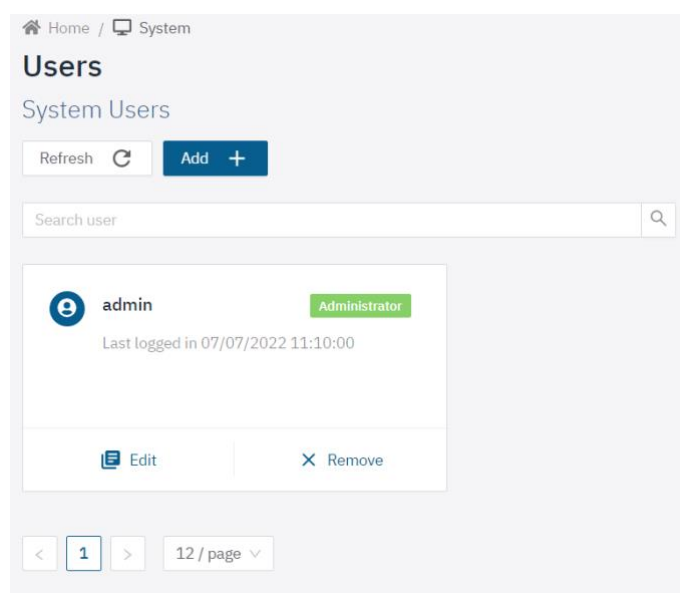
Figure 9 - Screen of System > Users menu

⚠️ **The factory default user and password must be modified for better access control and security**

Data e Hora

The Date and Time settings can be accessed through the *Equipment > Date and Time* menu and can be performed manually or through an NTP server.
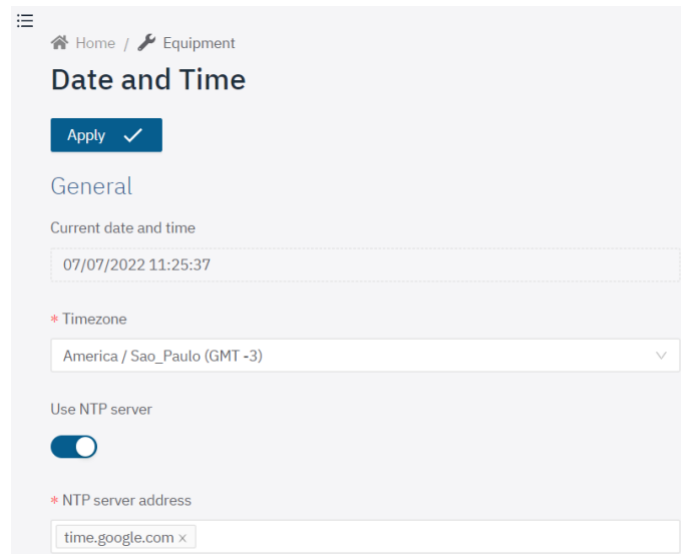
Figure 10 - Screen of Equipment > Date and Time menu
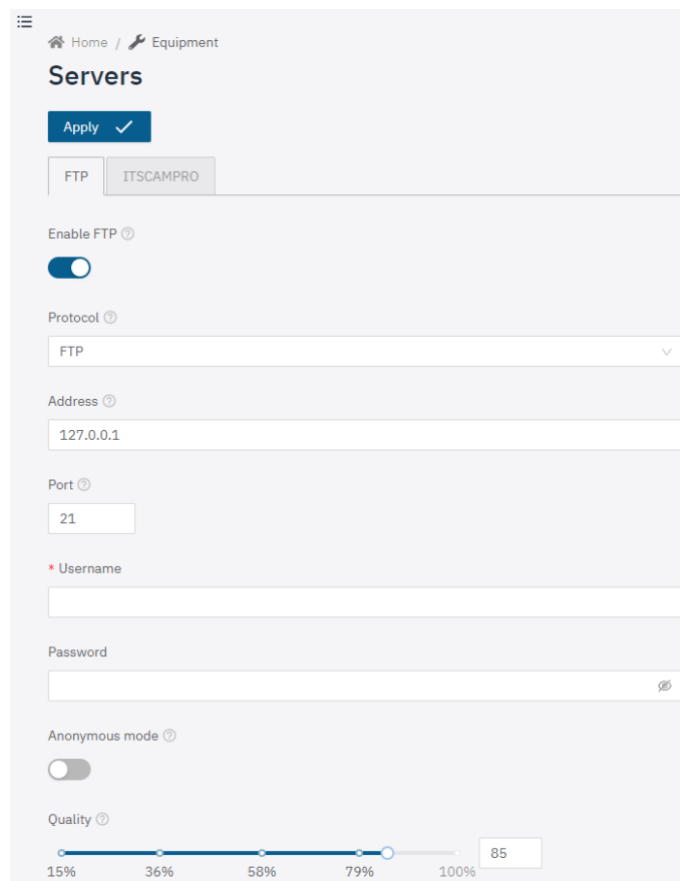
Servers

The ITSCAM 600 may send the images automatically to an FTP server or to the ITSCAMPRO.

1) Access the menu: *Equipment > Servers;*
2) Click on the tab corresponding to the server to be configured;
3) Check in Maintenance the steps for restoring the system to factory defaults in case any error occurs during the configuration of the servers.

FTP Servers

1) Enable the FTP server by clicking on *Enable FTP;*
2) Select the Protocol that will be used, from the options:
    a. FTP: Basic file transfer protocol;
    b. FTPS: Secure file transfer protocol by SSL/TLS;
    c. SFTP: File transfer protocol over SSH.
3) Fill in the IP *Address* in the Address field and the Access Port in the *Port* field;
4) Set a *User* and *Password* or enable *Incognito Mode* to not use user and password;
5) Check/Adjust the quality of the JPEG image when saved via FTP, in the *Quality* field;
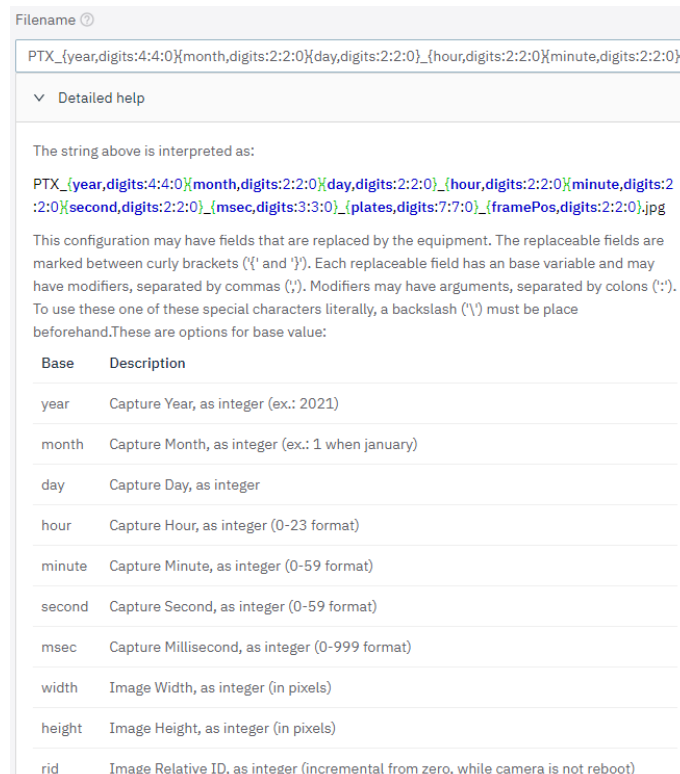
6) Change the code in *File Name* to customize the file name with capture information;

7) Enable the *Show Advanced Options* option and check/adjust the storage options that the FTP upload service will use;

8) Click on the *Apply* button after checking the entered data.

ITSCAMPRO Server

1) Click on the *ITSCAMPRO* tab;
2) Enable the *Enable ITSCAMPRO* option;
3) Fill in the IP *Address* in the Address field and the Access Port in the *Port* field;

4) Click on the *Apply* button after checking the entered data.

Plugins

The ITSCAM 600 allows the import of plugins directly through the web interface and the configuration of more than one port with external mapping.

Importing plugins requires an SD card formatted with ext4 to be inserted into the ITSCAM 600. In the absence of the SD card, the device will show an error message:
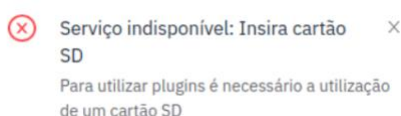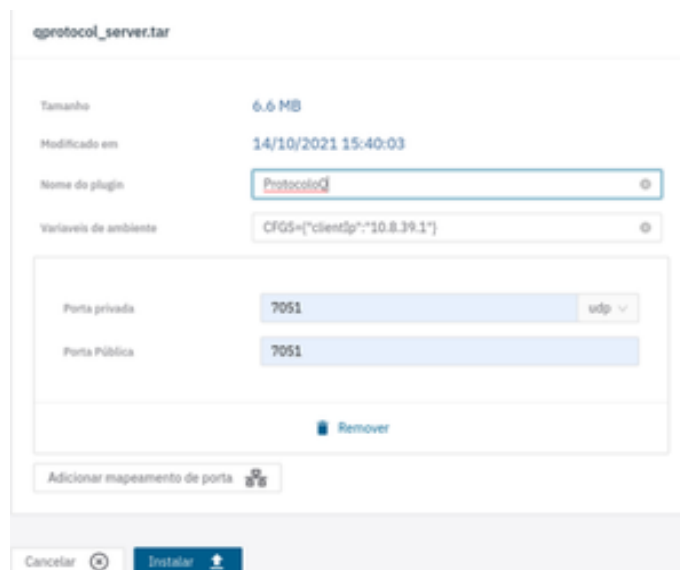
Figure 11 - Error message displayed in the absence of SD card

1) Access the menu: *System > Plugins;*

2) Click or drag a file in *.tar* format;
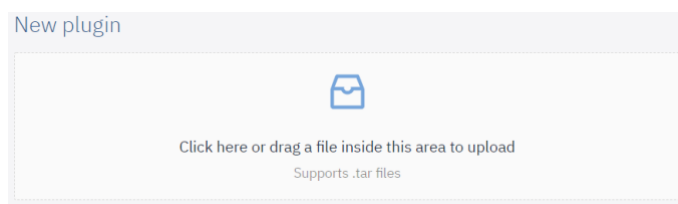
3) Set the *Plugin Name*;
4) Configure the plugin's Environment Variables*;*

5) Click on the *Install* button.

Image Profiles

The ITSCAM 600 has the *Day* and *Night* profiles of image capture configuration in the factory standard. The adjustments applied in each parameter can be accessed through the menu *Image > Profiles:*
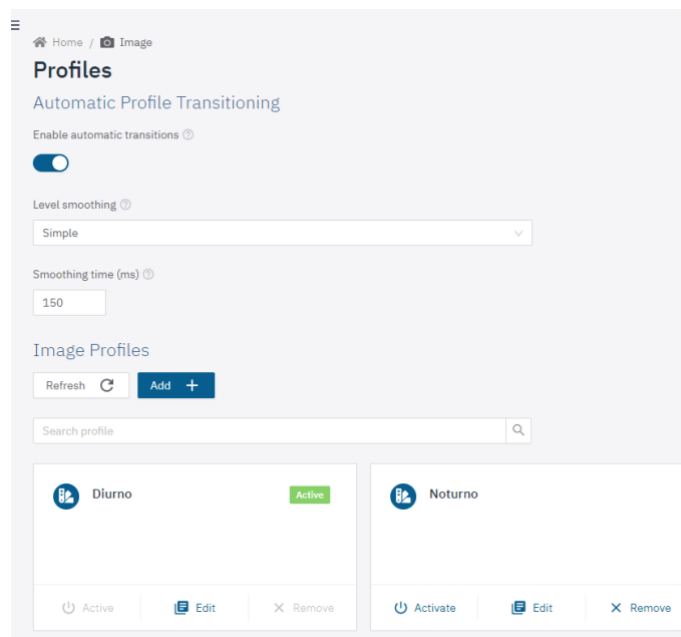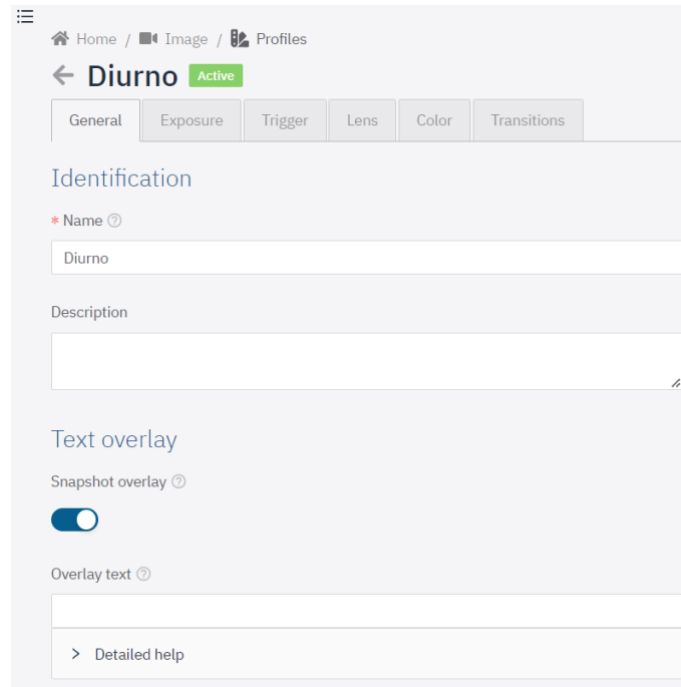
Figure 12 - Screen of Image > Profiles menu

1) Enable the *Enable automatic transitions* option, for automatic change between profiles according to internal settings and lighting conditions to obtain an optimal image quality in any situation.
    a. It is possible to register up to four *Image Profiles*, each with its own set of settings. It is recommended to use one profile for catches with ambient light (daytime) and another for catches with artificial lighting (nighttime).
2) Select *Level Smoothing;*
    a. *None:* change of the image settings performed instantly in the exchange of profiles;
    b. *Simple:* change of the image settings performed with a smoothing time in the exchange of profiles;
3) Set the *Time for Smoothing* in milliseconds;
4) Click on the *Add+* button under *Image Profiles*, to create new image profiles (it is recommended to configure existing profiles before creating new ones);
5) Locate the profile that will be configured;
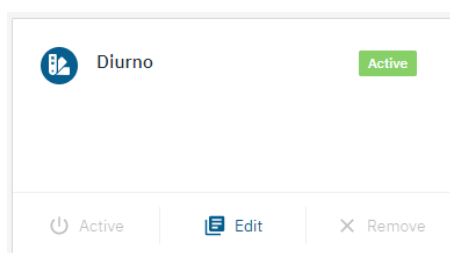
6) Click on *Edit;*

7) In this screen, the settings are automatically saved;
8) Identify the Profile by assigning a *Name*;
9) Enable the *Snapshot overlay* option, to apply a text caption to all photos;
10) Set the text overlay by filling in the *Overlay text* field in photos, with up to 1024 characters.

Lens

1) Click on the *Lens tab;*
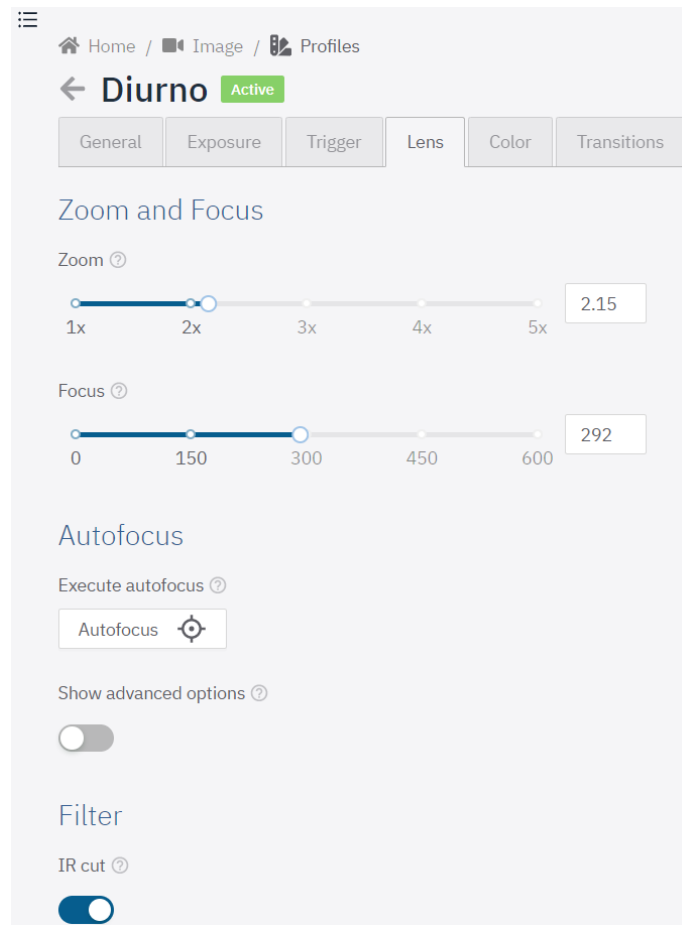
2) Check and adjust *Zoom* and *Focus* (this setting can also be performed through the [floating viewing window](#));

3) Perform the focus adjustment automatically by clicking on the *Autofocus* button [available only on models that have a motorized lens - ITSCAM 600 LM (S06L5AJP) and ITSCAM 600 LM AD IOT (S06L5W1AJP)];

4) Click on *Show Advanced Options* to display the advanced *Autofocus* settings;
   a. *Coarse step:* travels the entire focus range;
   b. *Contrast threshold:* the minimum contrast for the image to start being considered in focus;
   c. *Update rate:* expected time for the focus to be applied correctly to the image;
   d. *Region of interest:* defines the region in which *Autofocus* will be executed, the check considers by default only the central region of the image.

> ⚠️ **The adjustment of the advanced *Autofocus* options should only be performed in case of malfunction of the algorithm**

5) Enable the *IR Filter* when using infrared illuminators;
6) Adjust the positioning of the ITSCAM 600, based on the images displayed in the *Live Feed*. The suggested framework for two tracks in the image is the display of the ends of the road, as examples:

Figure 13 - Example of suggested framing with image generated in the daytime period

Figure 14 - Example of suggested framing with image generated during night time

⚠️ **Zoom and Focus Adjustment is linked to the Image Profile: Zoom and Focus adjustment must be performed *on all profiles* enabled in ITSCAM 600**

Image Framing

When the equipment is fixed using a suction cup on the windshield of a vehicle or on the ceiling, the image of the device will be upside down.

1) Access the menu: *Image > Miscellaneous;*
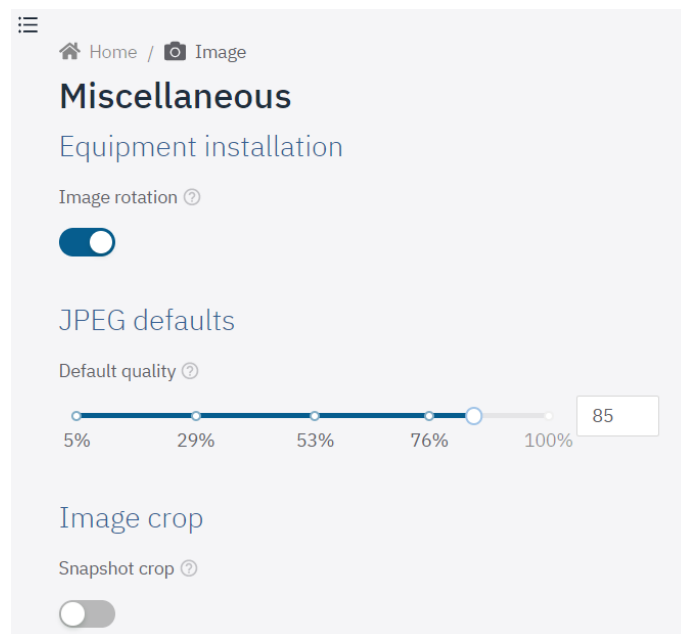
2) In this screen, the settings are automatically saved;
3) Enable the *Image Rotation* option;
4) Define the value of the *Default Quality* in which the JPEG will be generated;
5) Enable the *Image Crop* option, to generate the image using a specific area (this option does not affect streams and live image viewing);
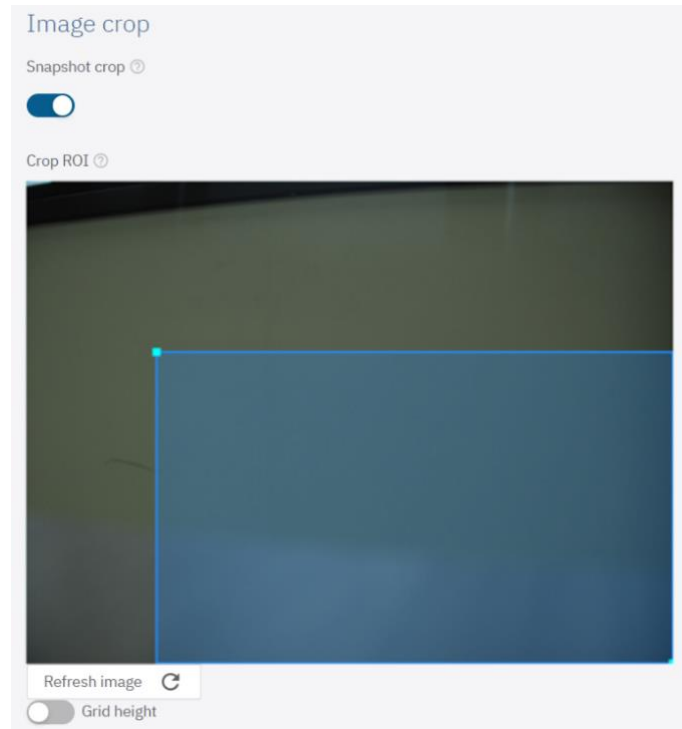
Live Feed

The ITSCAM 600 web interface has a live image viewer. It is available in a floating window, which can be moved to any location on the interface screen or expanded to occupy the entire screen.
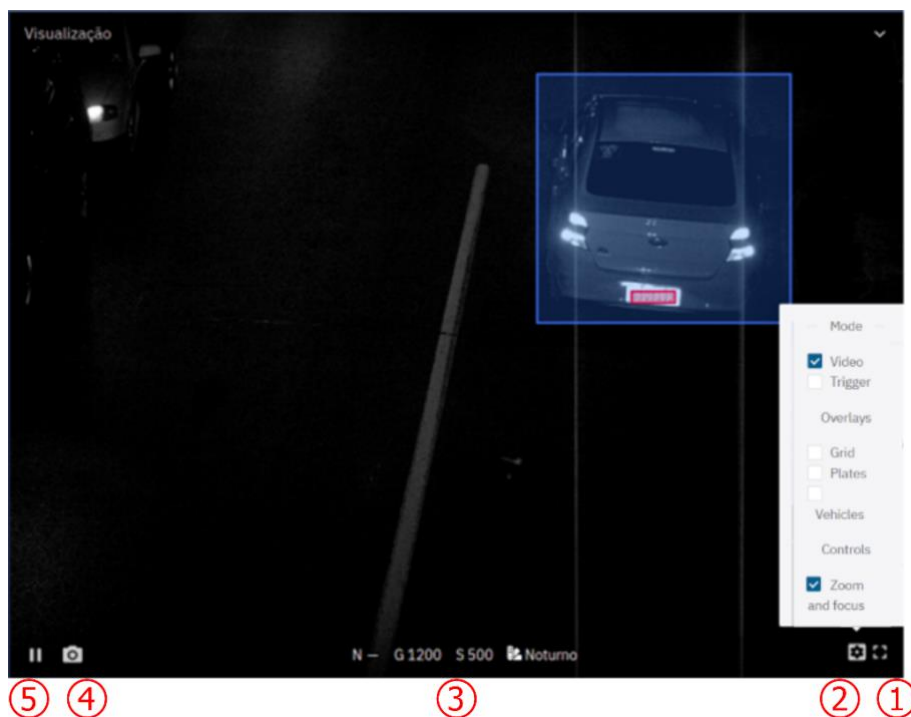
Figure 15 - Live Feed floating window: 1) Full Screen Button; 2) View settings button; 3) Level, Gain, Shutter values and Active Image Profile; 4) Take Photo Button; 5) Pause preview button

1) Click on *Full Screen* to expand the view and occupy the entire interface window;
2) Click on the *Live Feed* settings button to access the image preview options in the floating window:

3) *Mode*
    a. *Video:* displays the image that is being in real time by the device;
    b. *Capture:* displays the last captured image in the configured parameters.
4) *Overlays*
    a. *Grid:* displays a grid over the image;
    b. *License plates:* displays a recognition area on the license plates (as shown in the image);
    c. *Vehicles:* displays a recognition area on the vehicles (as shown in the image).
5) *Controls*
6) *Zoom and Focus:* displays two bars on the side of the floating window that allow you to adjust the Zoom and Focus of the image.
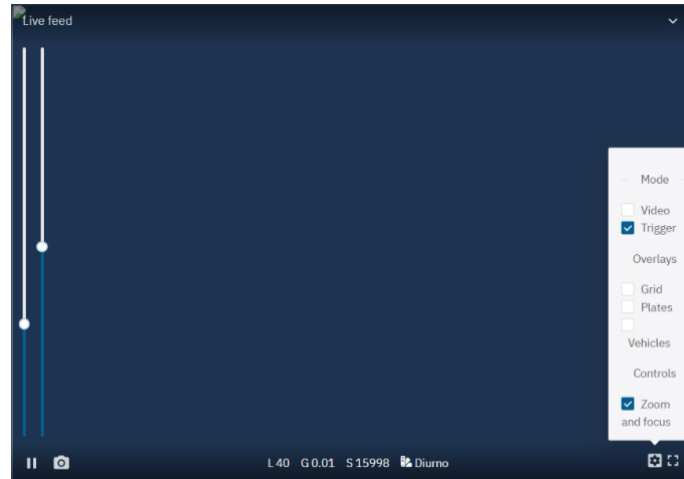
7) View, at the bottom of the floating window, the *Level, Gain and Shutter* values that are being applied on the device and the *Image Profile* that is active;
8) Click on the *Take Photo* button to download an instant JPEG image of the image being captured by the device;
9) Click on *Pause* preview to pause the live video and pin the last captured image.

Streams

The ITSCAM 600 has the option of following live the images captured by the device outside the web interface, integrated in a DVR or a video monitoring system.

Perform the stream settings by going to the *Video > Streams menu:*
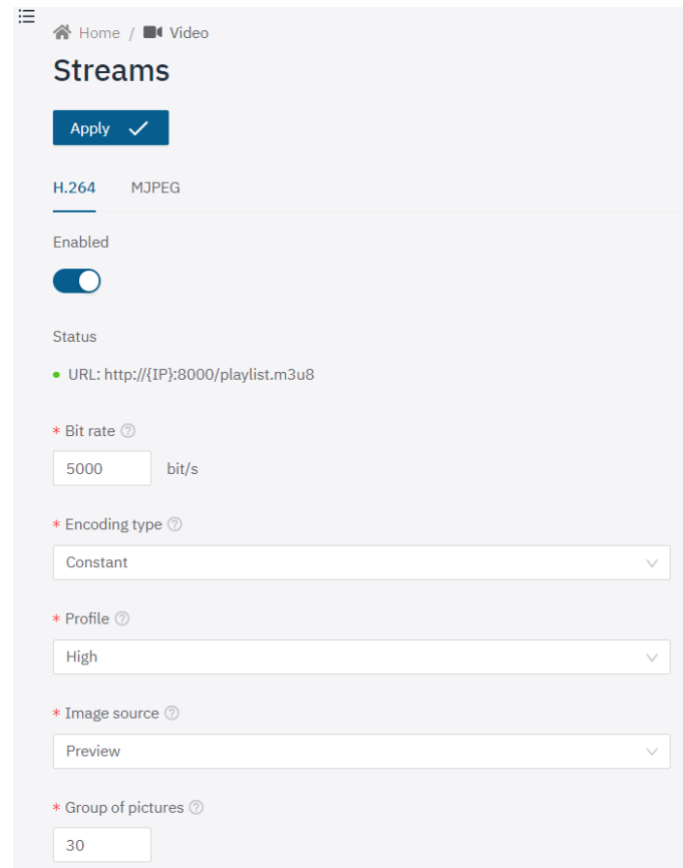
Figure 16 - Screen of Video > Streams menu

1) Enable the H.264/H.265 option by clicking on *Enabled*;
2) Choose the type of encoding from the options;
   a. H.264
   b. H.265
3) Set the data transfer *Bit rate* for the stream;
4) Select the *Encoding Type* from the options:
   a. *Constant:* the value remains the same throughout the video duration;
   b. *Variable:* the value decreases in parts that need less information and increases in parts that need a greater amount of data.
5) Select the *Profile* that defines the compression pattern (available only for H.264 encoding), among the options:
   a. *High*
   b. *Main*
   c. *Baseline*
6) Select *Image Source* from the options:
   a. *Preview*: using non-triggered images;
   b. *Stills*: using image triggers.
7) Click on the *MJPEG tab;*
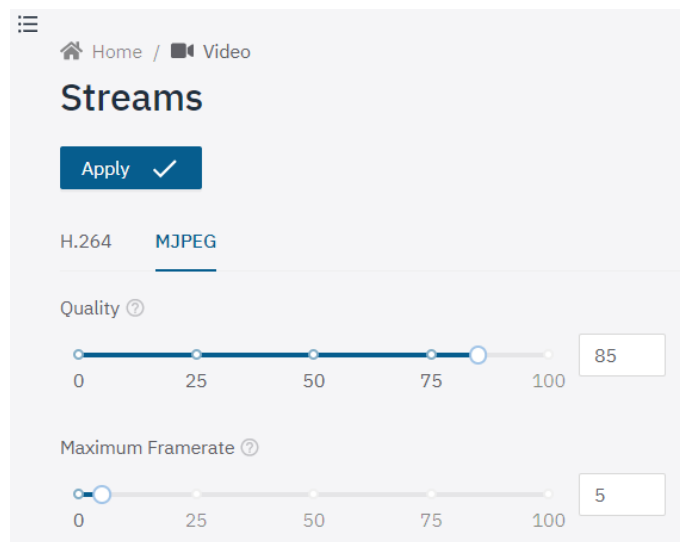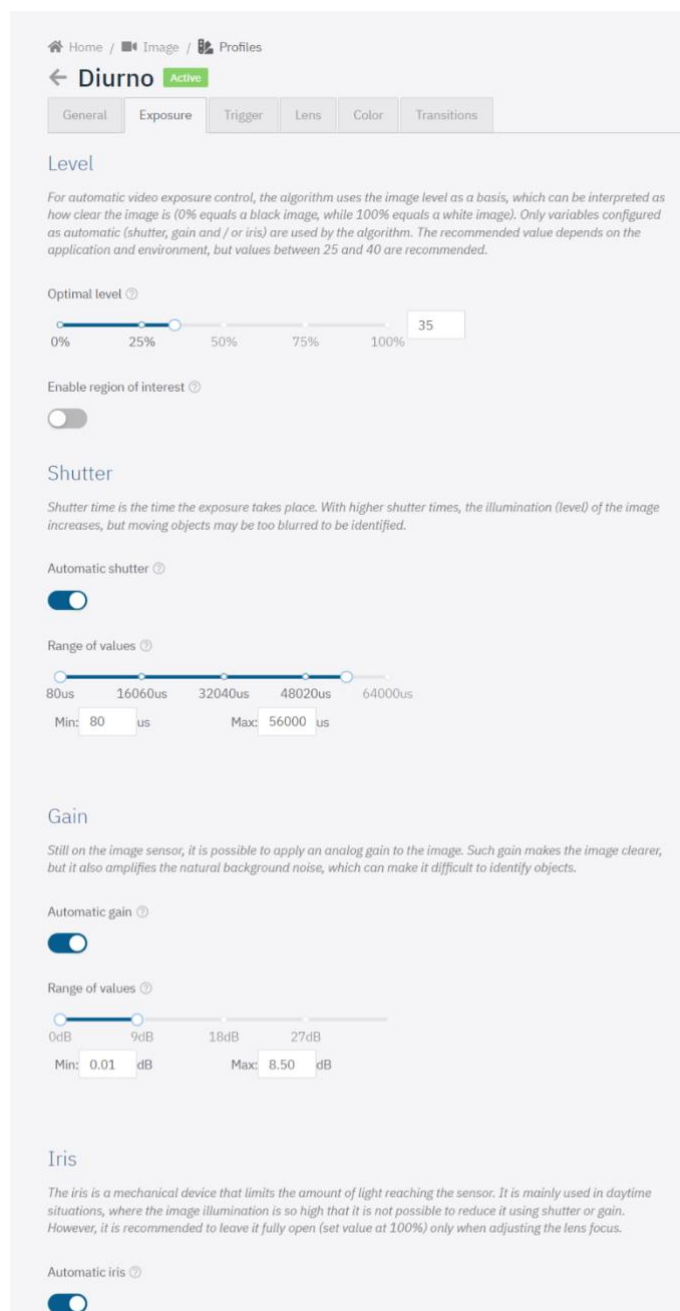
8) Adjust the quality of the images sent by the stream in the *Quality field;*
9) Set the maximum rate of images sent by the stream in the *Maximum Framerate* field;
10) Click on the *Apply* button after checking the information entered.

Exposure

1) Locate in the menu *Image > Profiles* the profile that will be configured;
2) Click on *Edit;*
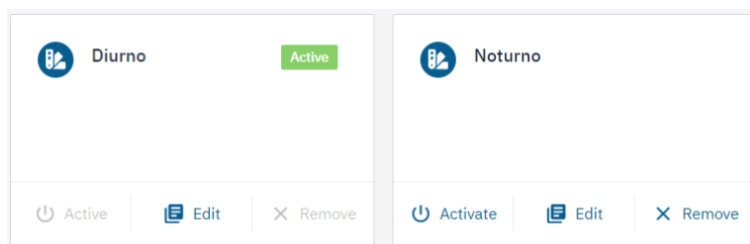
3) Click on the *Exposure* tab;
4) In this screen, the settings are automatically saved;

5) Select the target value of the clarity level that the ITSCAM 600 should attempt to achieve using the automatic controls at *Optimal Level*;
    a. Select the value of the *Level*, using values between 25 and 40;
6) select the *Enable Region of Interest* option and define which region of the image should be considered for level calculation, mainly in bright or very dark scenarios;
7) Enable the *Shutter* by clicking on the *Automatic Shutter* option;
    a. Select the values in microseconds in the *Range of Values* field;
8) Enable *Gain* by clicking on the *Automatic Gain* option;
    a. Select the values in decibel in the *Range of Values* field;
9) Enable *Iris* by clicking on the *Automatic Iris* option;

Color

It is not recommended to change the color settings, except for the *Gamma option.*

1) Locate in the menu *Image > Profiles* the profile that will be configured;
2) Click on *Edit;*
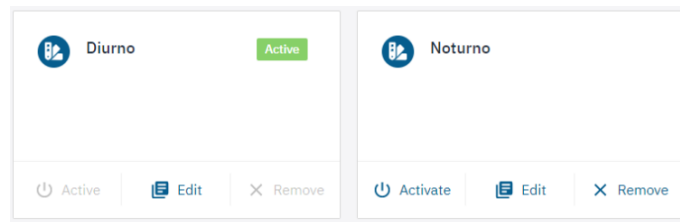
3) Click on the *Color tab;*
4) In this screen, the settings are automatically saved;

5) Set the Gamma values in *the Gamma* option;
6) Enable the *Automatic White Balance* option.

The values of *Brightness*, *Contrast*, *Saturation* and *Black Level* are factory adjusted to the ideal values under normal image lighting conditions and adjustments are restricted in cases of poor quality of the images generated. Contact Technical Support for guidance on these color settings
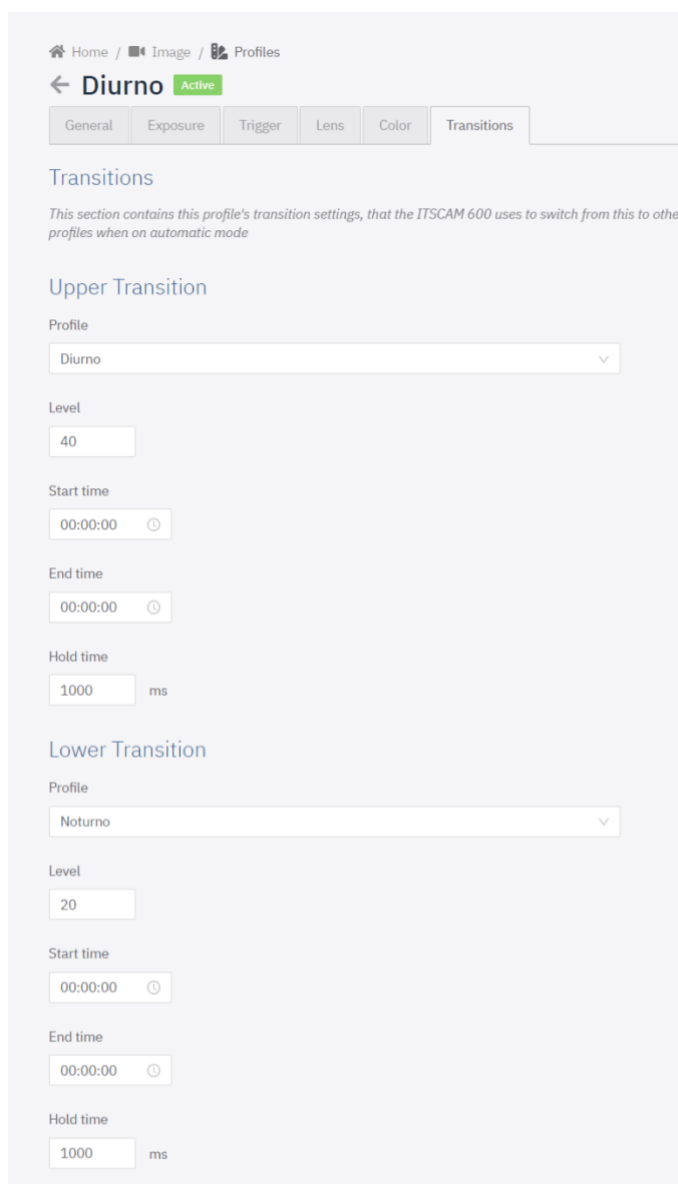
Profiles Transitions

When the Automatic Profile Transition option is enabled, the *Level* values and times are used as the basis for the transition.

1) Locate in the menu *Image > Profiles* the profile that will be configured;
2) Click on *Edit*;
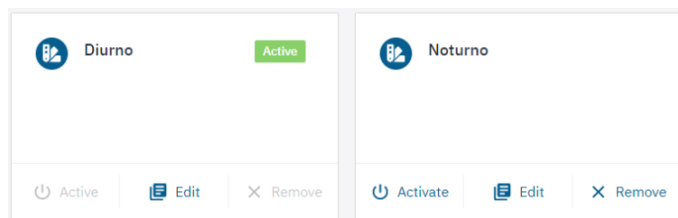
3) Click on the *Transitions* tab;
4) In this screen, the settings are automatically saved;

5) Set *Upper Transition* options:

    a. Choose the *Profile* that will be configured;
    b. Set the Image *Level* value;
    c. Choose the *Start Time*;
    d. Choose the *End Time*;
    e. Set the *Hold Time*;
6) Set the Lower Transition options:
    a. Choose the *Profile* that will be configured;
    b. Set the Image *Level* value;
    c. Choose the *Start Time*;
    d. Choose the *End Time*;
    e. Set the *Hold Time*;

Suggested Image Setting for Visible Light (Day Time Profile)

Capturing quality images is critical to realizing the full potential of the ITSCAM 600. Therefore, an image configuration suggestion is presented, which can be used as a starting point to adjust the equipment.

Parameter	Suggested Value
Level	40
Region of Interest	Disabled
Automatic Shutter	Enabled
Minimum Shutter	50
Maximum Shutter	1000
Auto Gain	Enabled
Minimum Gain	1
Maximum Gain	1500
Automatic Iris (on LM models)	Enabled
Iris (on LM models)	250
Automatic White Balance	Enabled
Multiple Exposures	Enabled
Exposure 1 - Flash power (using ITSLUX in installation)	100
Exposure 1 - Shutter	100
Exposure 1 - Use % of profile shutter	Enabled
Exposure 1 - Gain	100
Exposure 1 - Use % of Profile Gain	Enabled
Black level	20
Brightness	0
Contrast	0
Gamma	1
Saturation	0

Parameter	Suggested Value
Enable automatic transitions *	Checked
Profile (Upper Transition)	Daytime
Level (Upper Transition)	30
Hold time**	60,000
Profile (Lower Transition)	Nighttime
Level (Lower Transition)	10
Hold time**	60,000

*The configuration of the ITSCAM 600 configuration profiles shown is done in the *Transitions* tab at *Profile* menu.

**It is indicated to set *Hold Time* close to 1 minute (inserting in the interface the value of 60,000 ms) so that the ITSCAM 600 does not oscillate unnecessarily between the configuration profiles.

Suggested Image Setting for Infrared Light (Night Time Profile)

The nighttime images and/or captured by ITSCAM 600 with the aid of artificial infrared lighting need image configurations that highlight the details of the vehicles and the characters of the license plates. Therefore, an image configuration suggestion is presented, which can be used as a starting point for adjusting the equipment and checking the illuminator activation.

Parameter	Suggested Value
Level	40
Region of Interest	Disabled
Automatic Shutter	Enabled
Minimum Shutter	50
Maximum Shutter	500
Auto Gain	Enabled
Minimum Gain	1
Maximum Gain	1500
Automatic Iris (on LM models)	Enabled
Iris (on LM models)	1000
Automatic White Balance	Enabled
Multiple Exposures	Enabled
Exposure 1 - Flash power (using ITSLUX in installation)	100
Exposure 1 - Shutter	100
Exposure 1 - Use % of profile shutter	Enabled
Exposure 1 - Gain	100

Parameter	Suggested Value
Exposure 1 - Use % of Profile Gain	Enabled
Exposure 2 - Flash power (using ITSLUX in installation)	100
Exposure 2 - Shutter	100
Exposure 2 - Use % of profile shutter	Enabled
Exposure 2 - Gain	50
Exposure 2 - Use % of Profile Gain	Enabled
Black level	20
Brightness	0
Contrast	0
Gamma	25
Saturation	-100
Enable automatic transitions *	Checked
Profile (Upper Transition)	Daytime
Level (Upper Transition)	30
Hold time**	1,000
Profile (Lower Transition)	Nighttime
Level (Lower Transition)	10
Hold time**	1,000

Inputs and Outputs
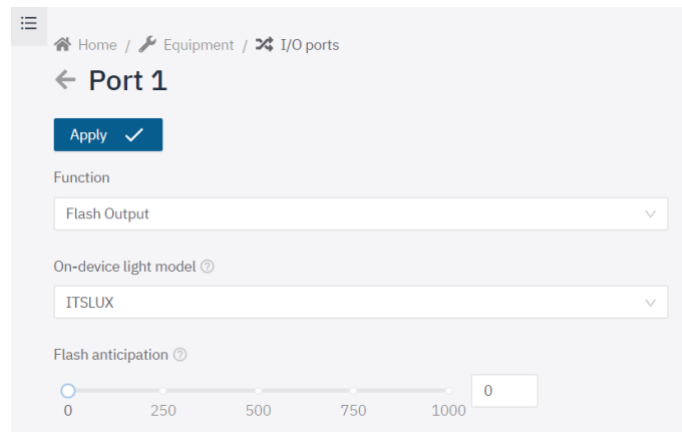
The ITSCAM 600 has 4 ports that must be configured by software as inputs or outputs (IOs), with IO1 and IO2 dedicated to the control of the illuminator activation and IO3 and IO4 available for the installation of external sensors, such as loops and light barriers, which identify the moment of image capture (trigger). To configure the use of each IO port:

1) Access the settings in the menu *Equipment > Inputs and Outputs;*
2) Locate the *Port* that will be configured;
3) Click on the respective *Edit* button;
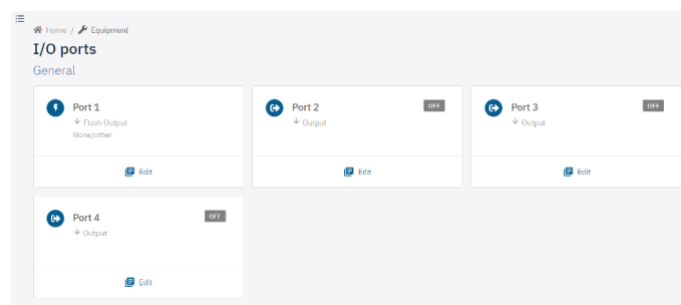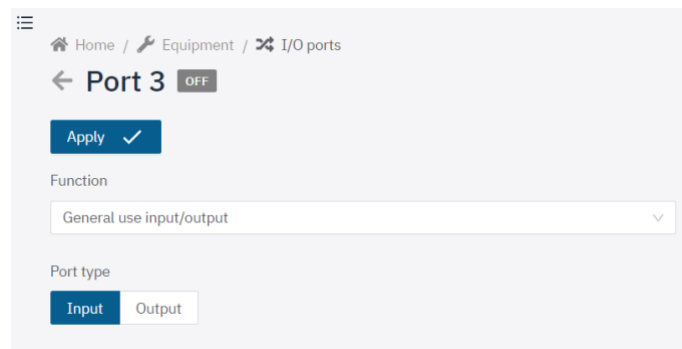
4) Select the *Function* as *Flash output* when the IO port is used for illuminator flash triggering (preferably ports 1 and 2):
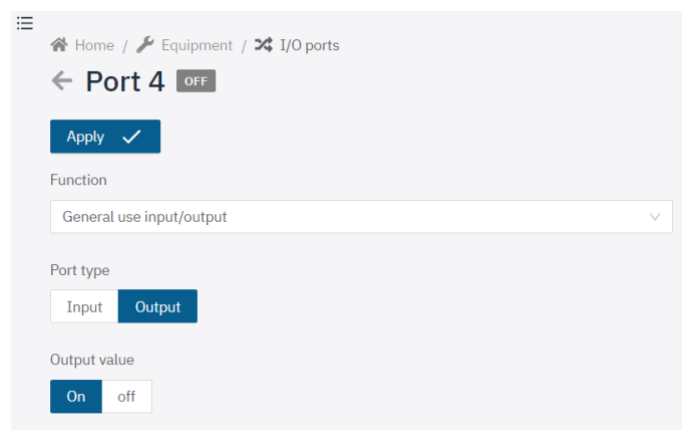
a. Select the *On-device light model*: *ITSLUX*, *WHITELUX*, *WHITELUX (video)* and, if the illuminator is from another manufacturer, *None/other;*
b. Set the *Flash anticipation* value, to align the image capture with the flash at its light emission peak, in equipment that has a delay in activation.

5) Select the *Function* as *General use Input/Output* preferably at ports 3 and 4:
   a. Select the *Port Type* as *Input* (example: when receiving a loop signal or light barrier);

b. Select the *Port Type* as *Output* (example: when triggering gates or alarms);
c. Select the *Output Value* as *On* to activate the output signal;
d. Select the *Output Value* as *Off* to disable the output signal;

6) Apply the port setting by clicking on the *Apply* button.

Trigger

1) Locate in the menu *Image > Profiles* the profile that will be configured;
2) Click on *Edit;*
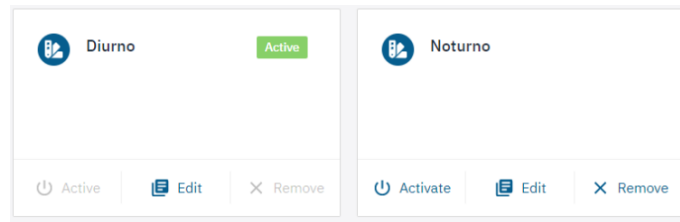
Revision 1.0

3) Click on the *Trigger tab;*
4) In this screen, the settings are automatically saved;
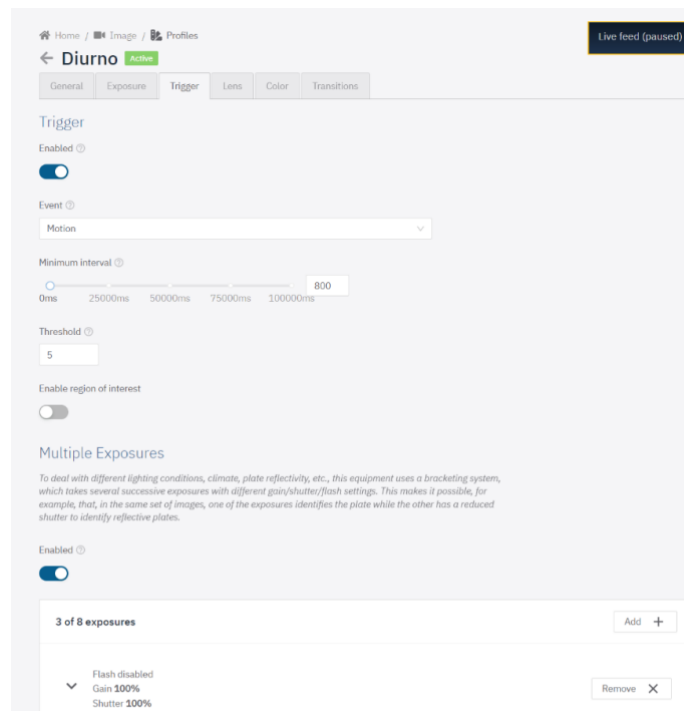
5) Select the *Enabled* option to configure *Trigger;*
6) Select the *Event* as *Constant* for the generation of a continuous stream of image capture, at the highest rate that the ITSCAM 600 can support;
7) Select the signal Input *Event* from the options:
    a. *Rising edge:* generates an image capture when the signal is activated;
    b. *Falling edge:* generates an image capture when the signal is deactivated;
    c. *Raising and falling edge:* generates an image capture when the signal is activated or deactivated;
    d. *High level:* generates image captures continuously while the signal is at high level;
    e. *Low level:* generates image captures continuously while the signal is at low level;
8) Select the Trigger *Event* as *Motion* to activate the *Virtual Trigger* functionality, which generates image captures while motion is detected;
9) Define the *Minimum interval* between triggers, in milliseconds;
10) Select the input *Port* of the signal used for the trigger, configured in *Inputs and Outputs;*
11) Set the *Threshold* value (*from 0 to 100*) required for the motion detector to be activated;
12) Select *Enable Region of Interest* to delimit the area of the image that should be considered when calculating the motion detector.

Conference of the Generated Images

1) Use the view of a capture to check the lighting, accessing the menu *Equipment > Images*, *Trigger* tab;
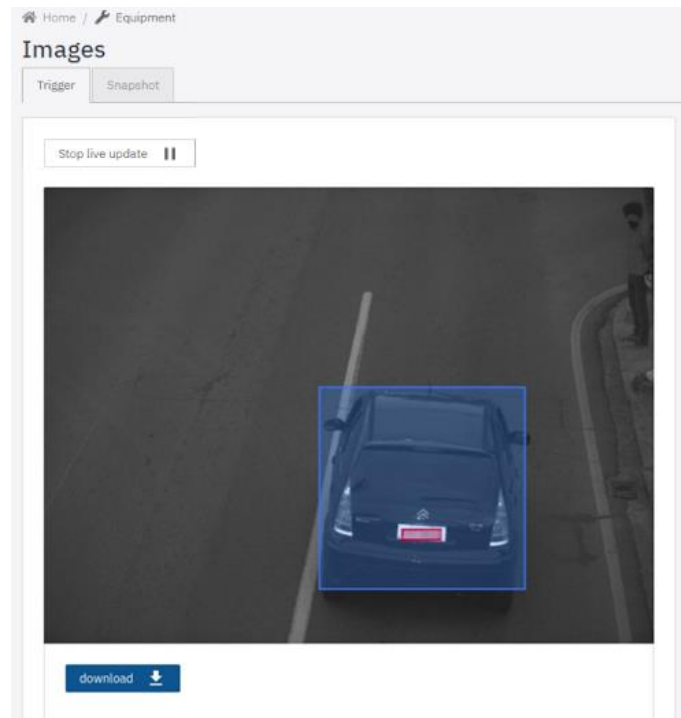
2) Click on the *Download* button to download an instant JPEG image of the image being captured by the device*;*

3) Click on the *Image Metadata* field to expand and view the generated image metadata;

4) Access the *Snapshot* tab to access the current settings applied to the image;

www.pumatronix.com

5) Click on *Take Photo* and view the *Metadata* of the generated image;
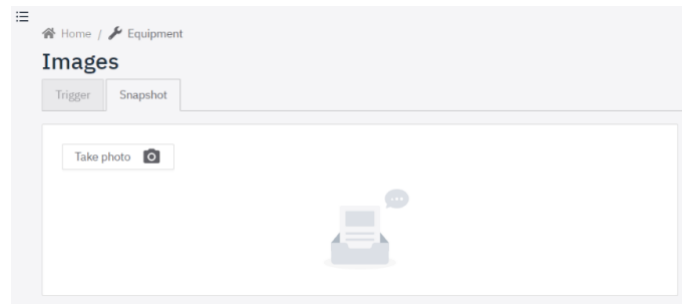
6) Make the necessary adjustments using the current image settings information.

Multiple Exposures

1) Locate in the menu *Image > Profiles* the profile that will be configured;
2) Click on *Edit;*
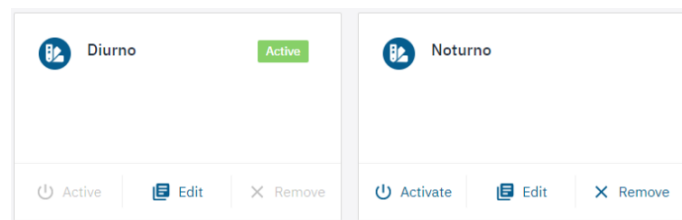
3) Click on the *Trigger tab;*
4) In this screen, the settings are automatically saved;
5) Locate *Multiple Exposures;*
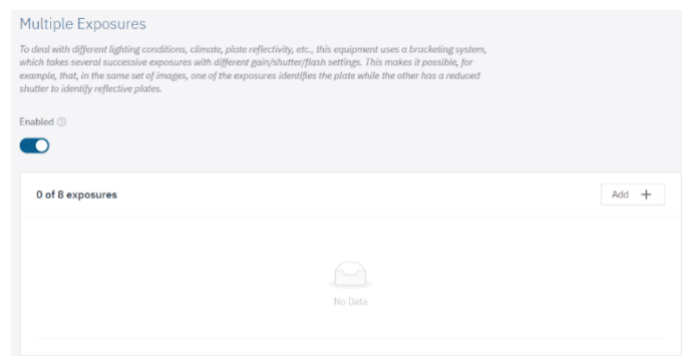
6) Enable *Multiple Exposures* by clicking on *Enabled*, to take multiple captures simultaneously with different *flash*, *shutter*, and *gain* parameters. If disabled, only one exposure is taken per trigger without flash and with shutter and video gain;
7) Click on the *Add +* button to create an exposure (up to 8 per profile);
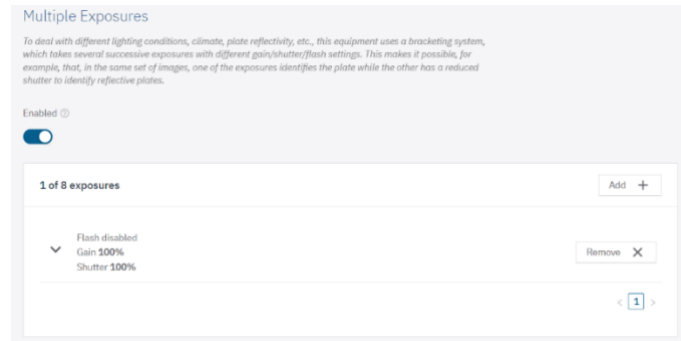
8) Click on the exposure to display the adjustments;
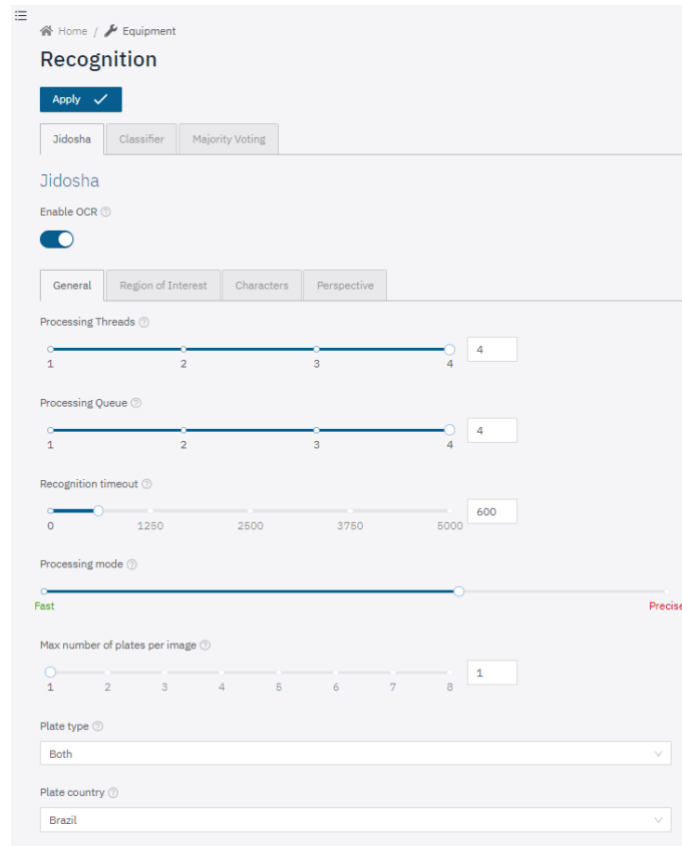
9) Select the *Flash Power*, always corresponding to a percentage of the initial shot (this option is available for the Pumatronix ITSLUX line of illuminators);

10) Select the *Shutter* percentage (exposure time of the image sensor), generating images with variation of the amount of light captured;

11) Select the percentage of *Gain* (digital post-processing), which allows to lighten or darken the images.

Recognition (OCR Reading)

The images captured by ITSCAM 600 provide automatic recognition of the license plate of the vehicles present in the images (OCR) and the type of vehicle identified (*Classifier*). In this way, ITSCAM 600 delivers JPEG images, with the comments field of the file containing the identified license plates, the type of vehicle identified (motorcycle, car, truck, and bus) and the instant configurations of the equipment.

1) Access the menu: *Equipment > Recognition*;
2) Click on the *Jidosha tab;*
3) Enable OCR by clicking on the *Enable OCR* option*;*
4) Click on the *General* tab;

a. Configure the number of *Processing Threads* to process the *OCR;*
b. Configure the maximum size of the OCR *Processing Queue* (smaller values decrease latency while larger values reduce frame loss in high flow situations);
c. Set the *Recognition Timeout* (images without plate or with hidden characters cause the algorithm to continue searching for the plate until the specified time is reached);
d. Set the *Processing Mode* (processing strategy used by OCR);
e. Set the *Maximum number of plates per image;*
f. Select the *Plate type* of the vehicles in the images (OCR searches the images for the pattern of letters and numbers of the license plate, but vehicles and motorcycles usually have different patterns);
g. Select the *Plate Country* of the vehicle license plate in the images;
   i. For OCR processing of vehicle plates from different countries in Brazil, please contact Pumatronix technical support. ITSCAM with on-board OCR performs processing for Brazilian license plates only.
5) Click on the *Region of Interest tab;*
   a. Enable the *Enable region of interest* option*;*
   b. Enable the *Grid* option (located just below the image of the Region of Interest);
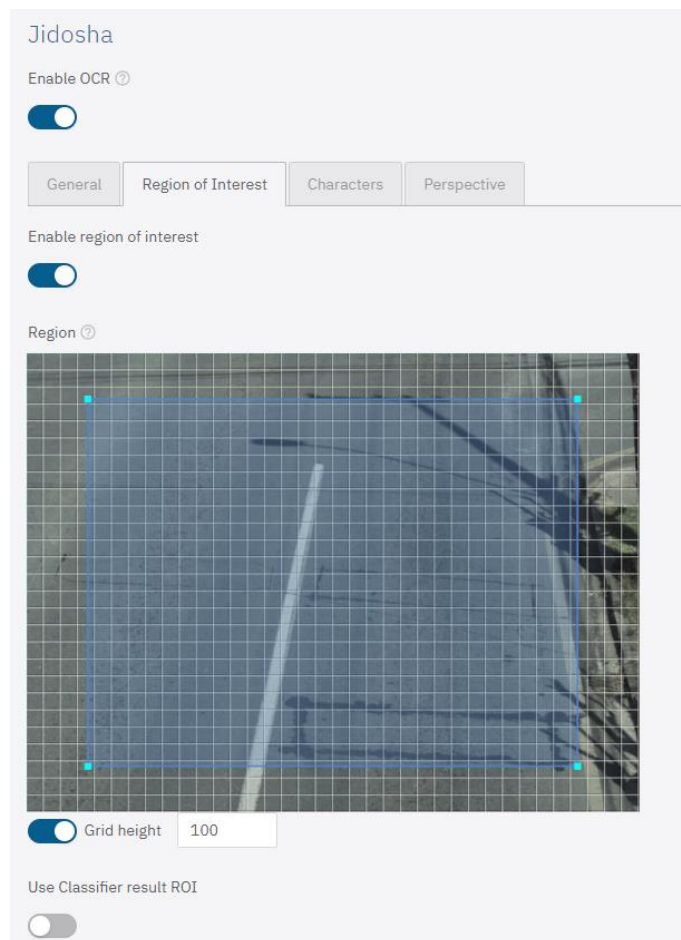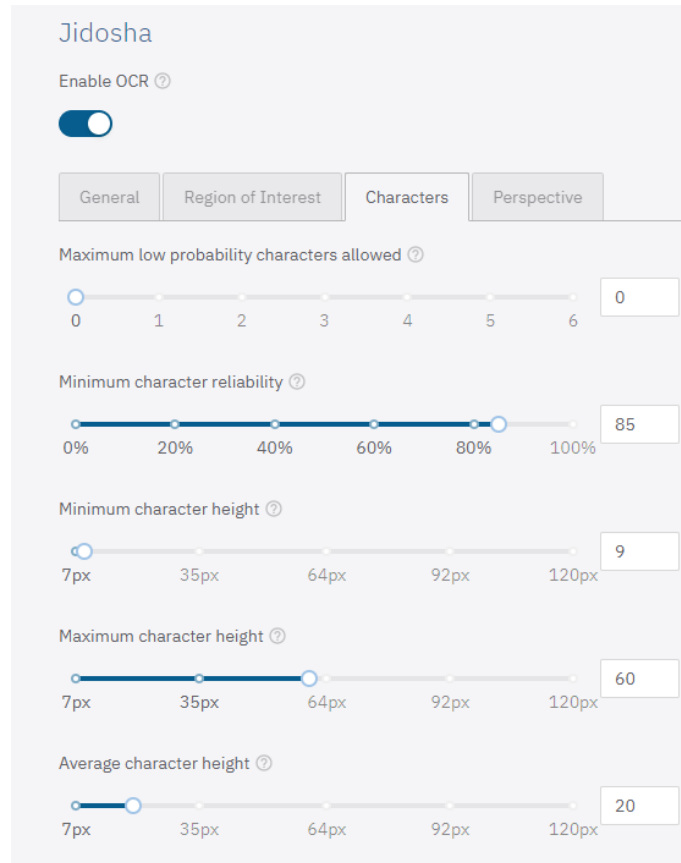
c. Set as 10 (for height of grid lines in pixels);

d. Note the size of the license plate characters in the Region marked as 1 in the image shown, that is, when the vehicles are at the location furthest from the capture device. If they are too different, you need to review the zoom adjustment.
e. Set as 40 (for height of grid lines in pixels);
f. Note the size of the license plate characters in the Region marked as 3 in the image shown, that is, when the vehicles are in the region closest to the capture device. Characters must be approximately the height of the grid. If they are too different, you need to review the zoom adjustment.
g. Set as 25 (for height of grid lines in pixels);

h. Note the size of the license plate characters in the Region marked as 2 in the image shown, that is, when the vehicles are in the central region of the image. Characters must be approximately the height of the grid. If they are too different, you need to review the zoom adjustment.

6) Click on the *Characters tab;*

Jidosha

Enable OCR ⑦

| General | Region of Interest | Characters | Perspective |

Maximum low probability characters allowed ⑦

0   1   2   3   4   5   6   `0`

Minimum character reliability ⑦

0%   20%   40%   60%   80%   100%   `85`

Minimum character height ⑦

7px   35px   64px   92px   120px   `9`
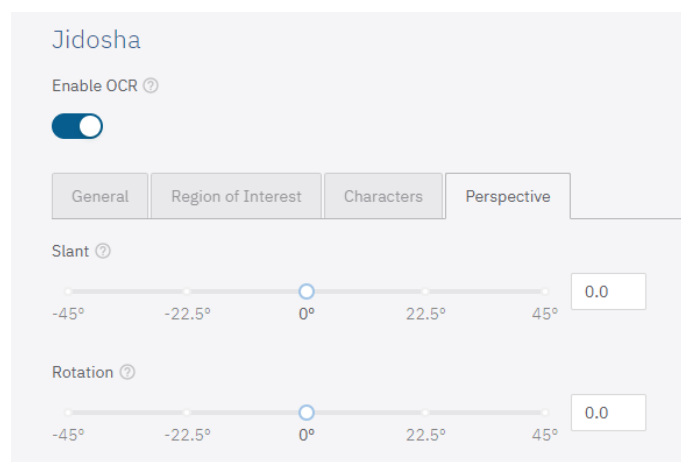
Maximum character height ⑦

7px   35px   64px   92px   120px   `60`

Average character height ⑦

7px   35px   64px   92px   120px   `20`

a. Set the *Maximum number of low probability characters allowed;*
b. Set the *Minimum Character Reliability;*
c. Specify as 9 the *Minimum Character Height;*
d. Specify as 60 the *Maximum Character Height;*
e. Specify as 20 the *Average Character Height;*

7) Click on the *Perspective tab;*

Jidosha

Enable OCR ⑦

| General | Region of Interest | Characters | Perspective |

Slant ⑦
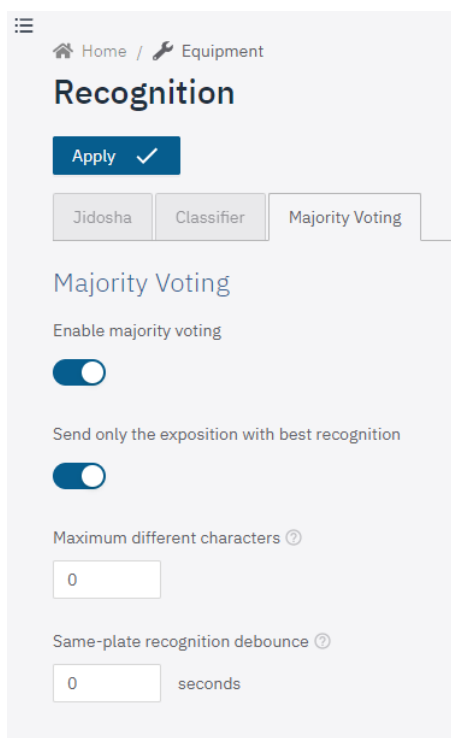
-45°   -22.5°   0°   22.5°   45°   `0.0`

Rotation ⑦

-45°   -22.5°   0°   22.5°   45°   `0.0`

a. Enter the *Tilt angle;*
b. Enter the *Rotation angle;*

8) Click on the *Majority Vote* tab;

a. Enable the *Enable majority vote* option for when the OCR is performed in *Multiple Exposures*, the resulting plate considers the result of the detection with greater reliability for each character;

b. Enable the *Send only the exposition with best recognition* option;

c. Set the value of the tolerated *Maximum different characters* to consider two plates to be equal;
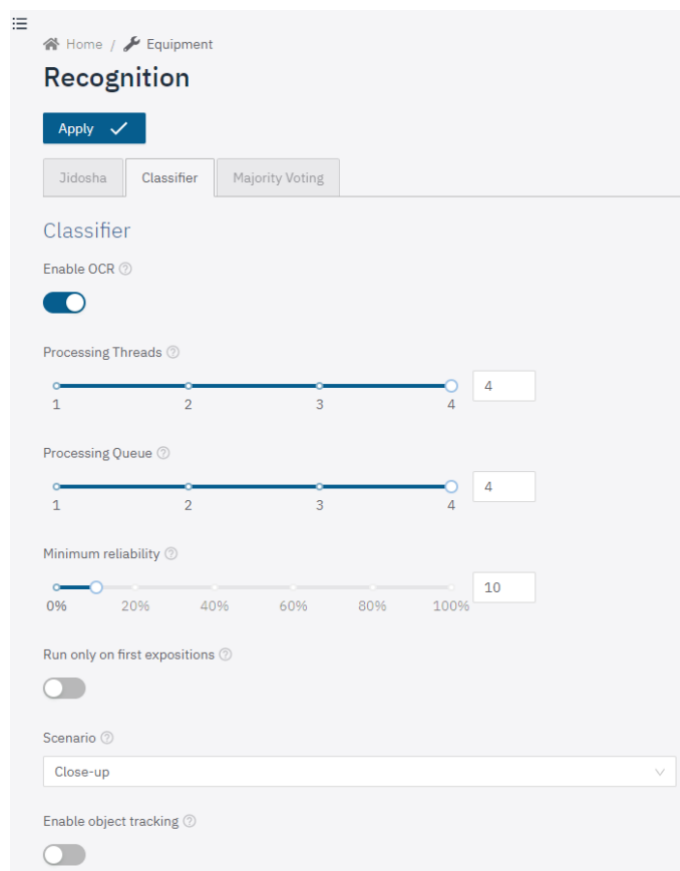
d. Set the value of the *Same-plate recognition debounce;*

9) Click on the *Apply* button when validating the information entered.

Classifier

1) Access the menu: *Equipment > Recognition*;
2) Click on the *Classifier tab;*

a. Enable the *Enable OCR* option for the device to analyze the captured images in real time and evaluate the content present in the images. This analysis aims to distinguish motorcycles, cars, trucks and buses from images that display only the track;
b. Configure the number of *Processing Threads;*
c. Configure the size of the Image *Processing Queue;*
d. Define the *Minimum Reliability* (for identification/classification of vehicles in the images);
e. Enable the *Run only on first expositions;*
3) Select *Scenario:*
a. *Panoramic* to capture two lanes;
b. *Close-up* for single lane;
4) Select the *Enable Object Tracking* option. In this function it is possible to create distinct trigger points for each lane and indicates the direction of the vehicle flow in order to generate a precise virtual trigger based on the class of objects;
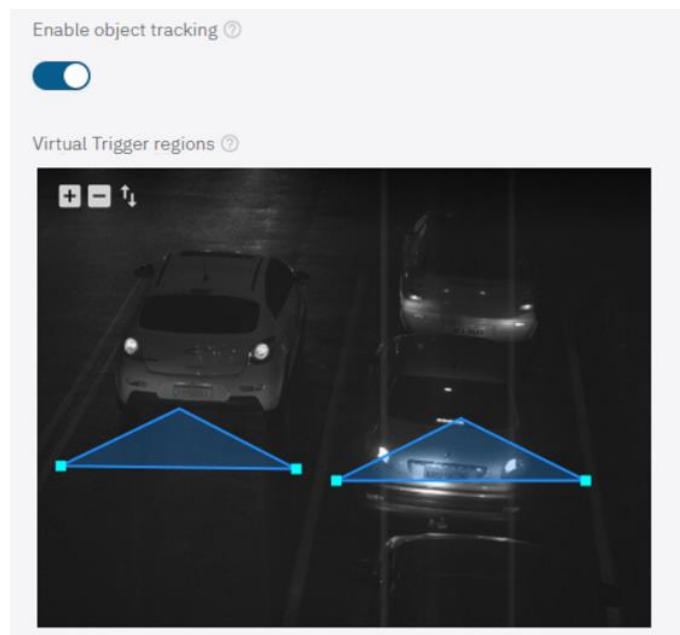
a. Click "+" to add trigger point;
b. Click "-" to remove trigger point;
c. Click "⇅" to change the direction of the Virtual Trigger region;

5) Click on the *Majority Vote tab*;
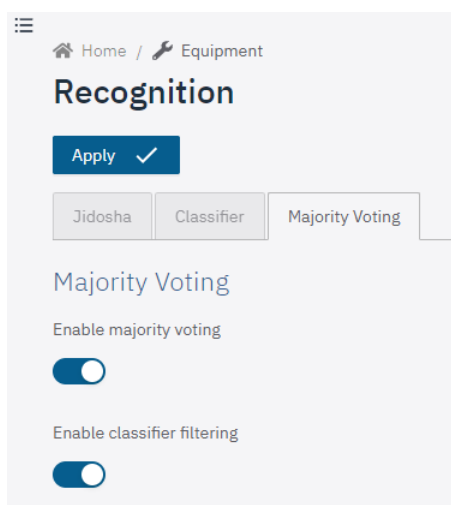
a. Enable the *Enable majority voting* option*;*
b. Enable the *Enable Classifier filtering* option so that only images that have a vehicle are generated;

6) Click on the *Apply* button after checking the information entered.

> ⚠️ **OCR in Region of Interest: OCR runs only within the drawn region, which is slightly shaded in the image.**

Licenses

The ITSCAM 600 allows the updating of the analytical licenses directly through the web interface, either for the automatic recognition of the license plate of the vehicles present in the images (OCR) or for the recognition of the type of vehicle identified (*Classifier*).

1) Access the menu *System > Licenses* to update the licenses;
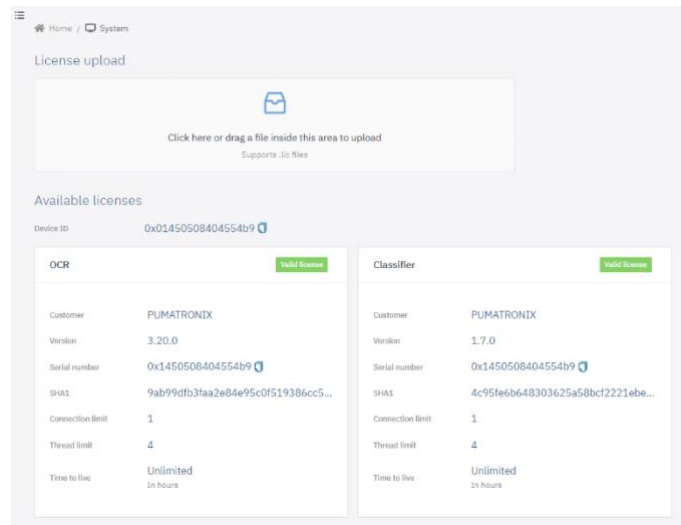
2) Click or drag the file in the *.lic* format that will be made available by Support or Commercial, when available.
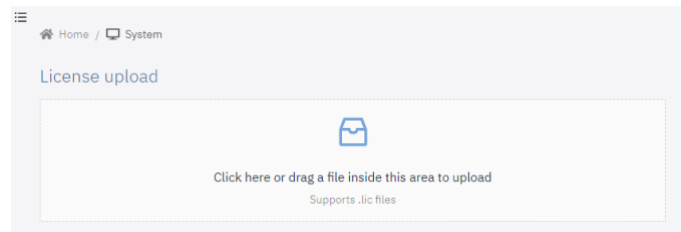
3) Please wait for the file to be uploaded.

Maintenance

1) Access the *Backup*, *Restore* and *Factory Defaults* maintenance options through the *System > Maintenance menu;*
2) Locate the *Backup* field to save a backup file of the settings, which can be restored on the device itself or to import the configuration to other devices;
3) Select the settings that will be saved in the backup file;
4) Click on the *Download Backup File* button;
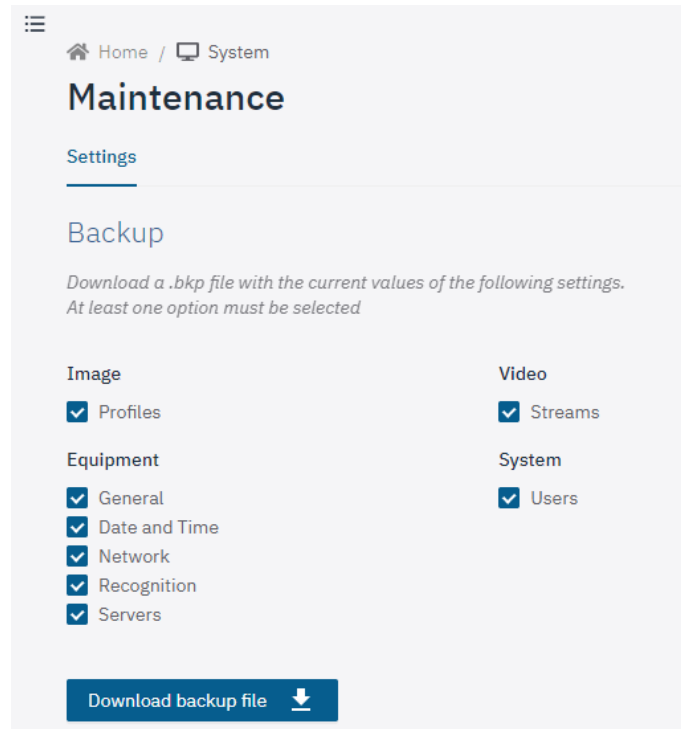
5) Locate the *Restore* field to use a backup, in which case the current ITSCAM 600 settings will be overwritten by the information saved in the file;
6) Click or drag a file in *.bkp* format;
7) Wait for the file to be uploaded and the equipment to restart.
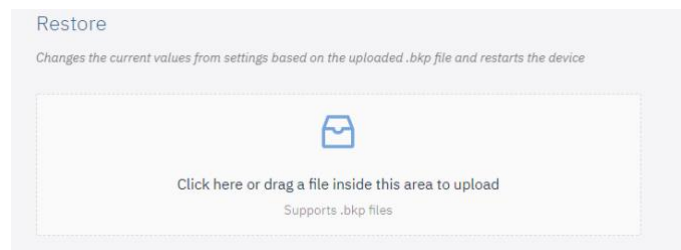
8) Locate the *Factory Defaults* field to restore Factory Defaults in case of malfunction or incorrect device settings;
9) Select the settings that will be reset to factory default;

Factory Defaults

Reverts the following settings to their default values and restarts the device.
At least one option must be selected

Image
☑ Profiles

Video
☑ Streams

Equipment
☑ General
☑ Date and Time
☑ Network
☑ Recognition
☑ Servers

System
☑ Users

Apply default settings ⟳

10) Click on the *Apply Default Settings* button;
11) Wait for the equipment to restart.

2.   API REST

The ITSCAM 600 has a API REST to access the equipment images and settings. This API follows the *OpenAPI* format and can be accessed through port 8081 (*http://IP_ITSCAM_600:8081/*).

The settings are available in the end-point scheme and can be accessed at *http://IP_ITSCAM_600:8081/v2/camera/profiles/0/schema.* This feature is only accessible if the firewall is disabled, and its use should be limited to the development environment.

Color Settings

Parameter	Minimum	Maximum
blacklevel	0	1023
brightness	-100	100
contrast	-100	100
gamma	0	255
saturation	-100	100

Exposure Settings

Gain		
Parameter	**Minimum**	**Maximum**
automatic	false	true
fixedValue	1	3600
maxValue	1	3600
minValue	1	3600

Iris		
Parameter	**Minimum**	**Maximum**
automatic	false	true
fixedValue	1	1000

Level		
Parameter	**Minimum**	**Maximum**
roi	false	true
roi: x0, x1, x2, x3	0	1632
roi: y0, y1, y2, y3	0	1220
target value	0	100

Shutter		
Parameter	**Minimum**	**Maximum**
automatic	false	true
fixedValue	53	64000
maxValue	53	64000
minValue	53	64000

Lens Settings

Parameter	**Minimum**	**Maximum**
exchanger	false	true
focus	0	600
zoom	0	2330

Motion Filter Settings (*MovFilter*)

Parameter	**Minimum**	**Maximum**
enabled	false	true
onlyCheck	false	true
roi	false	true
roi: x0, x1, x2, x3	0	1632
roi: y0, y1, y2, y3	0	1220

threshold	0	100

Multiple Exposure Settings

Parameter	Minimum	Maximum
enabled	false	true

Transitions Settings

Parameter	Minimum	Maximum
enabled	false	true
lower - endTime	0	20
lower - holdTime	100	100000
lower - level	0	10
lower - profile	0	2147483647
lower - startTime	0	20
lower - endTime	0	20
lower - holdTime	100	100000
lower - level	0	10
lower - profile	0	2147483647
lower - startTime	0	20

Trigger Settings

Parameter	Minimum	Maximum
enabled	false	true
event	0	99
minimumInterval	0	100000
port	1	4
roi	false	true
roi: x0, x1, x2, x3	0	1632
roi: y0, y1, y2, y3	0	1220
threshold	0	100

Whitebalance Settings

Parameter	Minimum	Maximum
automatic	false	true
red	0	800
green	0	800
blue	0	800

3. COUGAR Communication Protocol (Socket)

The Cougar protocol is an ITSCAM 600 integration interface based on TCP socket connections and the main functionalities made available through this API are: device control and settings; receiving images and metadata from vehicle ticket records. The protocol has the following features:

- Implementation Modularity
  - o All messages will share a common header, facilitating the implementation of the low-level interpreter;
  - o Use of the JSON format, allowing the automatic completion of all fields of some settings;
  - o Implementation of metadata and extra functionalities in the same command, without interfering with the operation;

- Asynchronous messages
  - o Metadata can be sent without interfering with the communication of settings;
  - o Facilitates the creation of GUIs or other high simultaneous event rate services;

- Connection customization
  - o Information is sent only when requested, reducing bandwidth consumption;
  - o Image metadata can be sent in isolation, allowing the receipt of only metadata, only the image or both;
  - o Full metadata is sent during JPEG processing, improving event timing and bandwidth usage;
  - o It allows the connection to use other binary JSON formats to reduce the amount of data transmitted;
  - o Increased security in connections with password usage request.

Connection and Messages

Cougar is implemented using TCP port 60,000. By default, the server will not send any data until it is configured to send or in response to a request, except for the server shutdown indication.

Every data sent will be encapsulated in a message and every message contains a header and a body. The data is formatted with the most significant byte (MSB) first, also called the *Network Byte Order*. The message structure is:

Cabeçalho											Corpo				
Byte Inicial	Tamanho (32bits)				Operação (16bits)		ID (32bits)				CRC (16bits)		Corpo			CRC corpo (16bits)
(102) 0x66	MSB			LSB	MSB	LSB	MSB			LSB	MSB	LSB	0	1	...	Tamanho-1	MSB	LSB
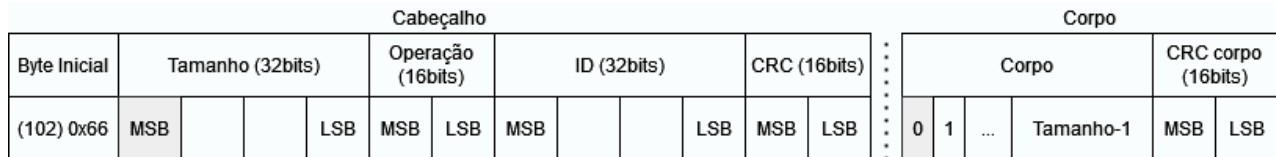
- Initial byte
  - 8 bits;
  - Fixed value of 102 (0x66);
- Body size
  - 32 bits without signal;
  - It contains the size of the body message (without CRC);
- Operation
  - 16 bits without signal;
  - It describes the operation performed or type of message on the body;
- ID
  - 32 bits without signal;
  - "Unique" transaction identifier (possible reuse of IDs with counting bearing);
  - Answers to requisitions have the same requisition ID.
  - Incremental value, with an increment of 2;
  - Client should start counting at 0. Server starts counting at 1;
- CRC
  - 16 bits without signal;
  - XMODEM format (polynomial: 0x1021, initial value: 0x0000, residue: 0x0000);
    - See calculation example of CRC16 XMODEM.
  - Calculation for shipment made from the initial byte to the ID (11 bytes);
  - It enables checking by running the algorithm only once in the entire header (13 bytes);
    - Header CRC will always return 0.
- Body (optional)
  - Size described in bytes in the header;
- Body CRC (when there is body)
  - 16 bits without signal;
  - XMODEM format (polynomial: 0x1021, initial value: 0x0000, residue: 0x0000);
  - Calculation done just for the body.

General Definitions

Important details of the implementation of ITSCAM 600:

- Equipment has 4 "dry I/Os" (usually called GPIO), which can be configured to:
  - Input of image capture signals (Trigger);
  - Output for equipment activation (gates, signaling, etc.);
  - Output for Flash activation;
- Equipment has 2 serial interfaces that can be mounted as RS-232 or RS-485 (usually being RS-232 on serial 1 and RS-485 on serial 2)
- Captured images can be separated into 3 categories:
  - Preview:

- o When there are no settings, the images are from Preview;
- o Main source of video streams;
- o No flash trigger;
- o No OCR processing or vehicle detection;
- o Used for motion trigger and image brightness adjustment;
- o *Snapshot*:
  - o Request made by the client (Cougar, WEB or Pumatronix Protocol);
  - o Multiple exposures (1 to 8): Flash, shutter and gain can be customized in the request;
  - o Processing pipeline where OCR and other analyses are done;
  - o Image is never filtered (there is always a response to the client);
- o *Trigger*:
  - o Request made automatically (via signal rise/fall edge, motion detection, etc.);
  - o Multiple exposures (1 to 8): Flash, shutter and gain previously customized via REST/WEB or Cougar;
  - o Processing pipeline where OCR and other analyses are done;
  - o Image can be filtered, if it is configured not to forward images without vehicles (majority vote);
- The equipment can take, from a single event, between 1 and 8 exposures, each with its own shutter and gain, which in turn can have a fixed value by the user or depend on the current value, and each can activate, with power control, up to 4 independent (flash) illuminators.

Operations

Type	Name	Value	Comment
General	NACK	1 (0x0001)	Message failure/ keep-alive
Events	SHUTDOWN	256 (0x0100)	Equipment/connection disconnecting
	EVT_TRIGGER	257 (0x0101)	Trigger Metadata
	JPEG_TRIGGER	258 (0x0102)	Trigger Image
	EVT_SNAPSHOT	259 (0x0103)	Snapshot Metadata
	JPEG_SNAPSHOT	260 (0x0104)	Snapshot Image
	EVT_PREVIEW	261 (0x0105)	Preview Metadata
	JPEG_PREVIEW	262 (0x0106)	Preview Image
	EVT_PIPE_START	263 (0x0107)	Image input in pipeline
	EVT_GPIO	264 (0x0108)	Change of GPIO inputs
	EVT_SERIAL	265 (0x0109)	Data received in the serial
Requests	SET_OPT_STR	512 (0x0200)	Section Setup
	SET_CALLBACKS	513 (0x0201)	Setup of callbacks (events)
	SET_JPEG_CFGS	514 (0x0202)	JPEG Conversion Settings
	TRIGGER_SNAPSHOT	515 (0x0203)	Request Snapshot
	GET_LASTFRAME	516 (0x0204)	Request last frame (Preview)
	AUTHENTICATE	517 (0x0205)	Client Authentication
	SET_SERIAL_CFGS	518 (0x0206)	Serial Interface Configuration

Type	Name	Value	Comment
	SEND_SERIAL_DATA	519 (0x0207)	Sending data through serial interfaces
	SET_EQUIP_CFGS	520 (0x0208)	General equipment settings

The body of all messages has one of the following formatting:

- None: (empty body [0 bytes])
  - Used for NACK as a check-alive (to check more quickly if the connection to the server has dropped);
- Binary: Body contains only data in binary format (i.e.: JPEG image)
  - Used for GET_LASTFRAME;
- JSON: Body contains data in JSON format only or one of the binary variants, configured using SET_OPT_STR;
  - Used in all other requests and events without JPEG image (i.e.: EVT_SNAPSHOT);
- Mixed: Body has, in order:
  - Metadata size in 32 bits (4 bytes);
    - MSB first, as well as for the header.
  - Metadata in JSON (or variant) format;
  - Data in binary format, occupying the rest of the space specified by the header;
  - Used in image events (i.e.: JPEG_SNAPSHOT);

JSON field descriptions accompany, in square brackets, the data type along with the default value or an example. For example:

- "field" [string, "value"]: Explanation of the field.

For most fields, the response contains all the fields that can be configured for a given request. Therefore, to read the current settings, send an empty requisition.

NACK

NACK is sent by the server in response to a command when:

- Response to NACK command (sent by customer)
  - Because the TCP connection often assumes that the server is still operational, using a check-alive can verify that the server has shut down faster.
- Invalid operation or not implemented by the server;
- Client not authenticated (if the option is enabled);
- Required operand is not present or is incorrectly formatted;
- Internal failure.

A NACK is not sent when:

- Message header is malformed, with invalid header or body CRC;
- Non-critical operation fails or is incorrectly formatted
  - i.e.: when trying to set the "trigger" field in the SET_CALLBACKS command to 1, the operation fails because it expects a Boolean value (true or false) but does not generate errors. You need to check the command response to determine if the operation was successful.

The response body is a JSON with the "reason" field that describes the error found for debugging, unless it is a response to the NACK command itself (in this case the body is empty).

SHUTDOWN

Event sent when the server is shutting down (for example, if the equipment is rebooting). Sent without arguments and does not need to be configured to be sent.

SET_OPT_STR

SET_OPT_STR configures options related to the client section/connection. For security, this is the only configuration that operates only with the simple JSON format (e.g. {"key":"value"}). The settings for this option are:

- "json" *[string, "plain"]*: mode of sending all other messages. It can be:
  - "plain": Common JSON (string ascii);
  - "bson": Binary JSON (BSON);
  - "cbor": Concise Binary Object Representation (CBOR);
  - "messagepack": MessagePack;
  - "ubjson": Universal Binary JSON;
- "respondCfgPath" *[bool, false]*: When set, the SET_EQUIP_CFGS operation returns the path that was sent to the request (more details on the operation).

AUTHENTICATE

If the server is configured as such, the client will need to send a password for authentication. This option is highly recommended for any operating system since Cougar has access to most configurations of the equipment. The parameter used in the request (which can be omitted if you only want to check the status) is:

- "pass" *[string]*: Password for customer access.

This command does not return NACK. The answer will always contain:

- "auth" *[bool]*: True if access to other messages is released;
- "msg" *[string]*: Message related to the status of the operation. It can indicate, for example:
  - If authentication is not required;
  - If it is already authenticated;
  - Whether the authentication has been successful;
  - If the field is malformed;
  - If the password is incorrect;

The only commands that work without needing authentication are NACK (check-alive), SET_OPT_STR (to set JSON mode) and AUTHENTICATE.

SET_CALLBACKS

SET_CALLBACKS configures which events are sent to the client. The fields are as follows:

- "pipeline" *[bool, false]*: Enables EVT_PIPE_start events;
- "trigger" *[bool, false]*: Enables EVT_TRIGGER events;
- "snapshot" *[bool, false]*: Enables EVT_SNAPSHOT events;
- "preview" *[bool, false]*: Enables EVT_PREVIEW events;
- "gpio" *[bool, false]*: Enables EVT_GPIO events;
- "triggerjpeg" *[bool, false]*: Enables JPEG_TRIGGER events;
- "snapshotjpeg" *[bool, false]*: Enables JPEG_SNAPSHOT events;

Revision 1.0

- "previewjpeg" *[bool, false]*: Enables JPEG_PREVIEW events;
- "serial1" *[bool, false]*: Enables EVT_SERIAL events for serial 1;
- "serial2" *[bool, false]*: Enables EVT_SERIAL events for serial 2.

SET_JPEG_CFGS

SET_JPEG_CFGS configures quality and minimum timing of JPEG generation. The default quality is configurable via REST/WEB interface. The fields are as follows:

- "trigger"
    - "quality" *[int, configurable]*: Quality of images generated from Trigger.
- "snapshot"
    - "quality" *[int, configurable]*: Quality of images generated from Snapshot.
- "preview"
    - "quality" *[int, configurable]*: Quality of images generated from Preview;
    - "mindt" *[int, 100]*: Minimum time (in milliseconds) between Preview images, to reduce the maximum frame rate.

EVT_PIPE_START

Event sent when a frame enters the photo pipeline (Trigger and Snapshot). Used for easy timing of taking photos with an external server. Only the simplest metadata is sent:

- "framecount" *[uint64]*: Single image value, implemented using a counter that increments with all images taken (Snapshots, Triggers and/or Previews), zeroed at server startup;
- "rid" *[uint64]*: Snapshot sync value. Implemented so that the customer can correlate the request made (which returns the same value) with the image taken
- "multexp"
    - "len" *[int]*: Number of exposures in the multi-exposure group;
    - "pos" *[int]*: Position of the image at multiple exposures, starting with 0.

EVT_TRIGGER, EVT_SNAPSHOT and EVT_PREVIEW

All these events are sent when the images have just been processed internally and are delivered to the Server. The moment these events are sent, the image JPEG conversion also starts, if enabled. For Trigger and Snapshot images, it is possible to correlate EVT_PIPE_START events using "framecount". The metadata of all of them are similar, except for the lack of "rid", "jidosha" and "classifier" data for Preview images. The fields are as follows:

- "framecount" *[uint64]*: Single image value, implemented using a counter that increments with all images taken (Snapshots, Triggers and/or Previews), zeroed at server startup
- "rid" *[uint64]*: Snapshot sync value. Implemented so that the customer can correlate the request made (which returns the same value) with the image taken;
- "ogSize"
    - "w" *[int]*: Original image width, in pixels;
    - "h" *[int]*: Original image height, in pixels;
- "size"
    - "w" *[int]*: Image width. Unlike the original if the image has been cropped, in pixels;
    - "h" *[int]*: Image height. Unlike the original if the image has been cropped, in pixels;
- "multExp"
    - "len" *[int]*: Number of exposures in the multi-exposure group;
    - "pos" *[int]*: Position of the image at multiple exposures, starting with 0;

- "shutter" *[int]*: Shutter of exposure in microseconds
- "gain" *[int]*: Analog gain from exposure in millibels (100x decibels)
- "stats"
  - o "level" *[int]*: Image level, with a value from 0 to 1000;
  - o "meanr" *[int]*: Average value of the color Red, with a value from 0 to 255;
  - o "meang" *[int]*: Average value of the color Green, with a value from 0 to 255;
  - o "meanb" *[int]*: Average value of the color Blue, with a value from 0 to 255;
  - o "stddev" *[int]*: Standard deviation squared, with a value from 0 to 65535;
- "time"
  - o "setup" *[uint64]*: Monotonic exposure scheduling time, in microseconds;
  - o "exp" *[uint64]*: Monotonic exposure time, in microseconds;
  - o "dma" *[uint64]*: Monotonic time of exposure entry into the pipeline, in microseconds;
  - o "now" *[uint64]*: Internal monotonic time of this message, in microseconds;
- "date"
  - o "year" *[int]*: Year the image was shot;
  - o "month" *[int]*: Month the image was shot;
  - o "day" *[int]*: Day the image was shot;
  - o "hour" *[int]*: Time the image was shot;
  - o "min" *[int]*: Minute the image was shot;
  - o "sec" *[int]*: Second the image was shot;
  - o "msec" *[int]*: Millisecond the image was shot (taken at the same time as the "time.dma" field);
- "crop"
  - o "x" *[int]*: Horizontal position in the beginning of the image cropping, in pixels;
  - o "y" *[int]*: Vertical position in the beginning of the image cropping, in pixels;
- "jidosha"
  - o List / Vector
    - "plate" *[string]*: Plate identified;
    - "probs" *[float list]*: Probabilities of each identified character. Same size plate identified. Values from 0.0 to 1.0;
    - "pos"
      - "x" *[int]*: Horizontal position of the plate, in pixels;
      - "y" *[int]*: Vertical position of the plate, in pixels;
      - "w" *[int]*: Plate width, in pixels;
      - "h" *[int]*: Plate height, in pixels;
    - "color" *[int]*: Color of the plate, with 0 for white and 1 for red;
    - "moto" *[int]*: Value 1 for when the identified plate has a motorcycle's shape, 0 if not;
    - "country" *[int]*: OCR country code, using ISO 3166-1;
- "classifier"
  - o List / Vector
    - "type" *[int]*: Identified vehicle type, based on the Classifier interface:
      - 0: Unknown;
      - 1: Car;
      - 2: Motorcycle;
      - 3: Truck;
      - 4: Bus;
    - "prob" *[float]*: Probability of identification, from 0.0 to 1.0
    - "pos"
      - "x" *[int]*: Horizontal position of the vehicle, in pixels;
      - "y" *[int]*: Vertical position of the vehicle, in pixels;
      - "w" *[int]*: Vehicle width, in pixels;
      - "h" *[int]*: Vehicle height, in pixels.

JPEG_TRIGGER, JPEG_SNAPSHOT and JPEG_PREVIEW

All these events load the converted JPEG image along with some metadata related to the capture, so that the correlation of events can be done (using the Mixed formatting, previously described). The quality of the conversion is configured via "quality" fields of SET_JPEG_CFGS. For the JPEG_PREVIEW event, only the "framecount" and "quality" fields are populated. For others, the fields are:

- "framecount" *[uint64]*: Single image value, implemented using a counter that increments with all images taken (Snapshots, Triggers and/or Previews), zeroed at server startup
- "quality" *[int]*: JPEG conversion quality, in percentage;
- "rid" *[uint64]*: Snapshot sync value. Implemented so that the customer can correlate the request made (which returns the same value) with the image taken;
- "multExp"
    - "len" *[int]*: Number of exposures in the multi-exposure group;
    - "pos" *[int]*: Position of the image at multiple exposures, starting with 0.

TRIGGER_SNAPSHOT

This operation queues a Snapshot request. By default, when called without any parameters, it uses the default parameters configured for Trigger in the WEB/REST interface. Multiple exposure settings should be formatted as follows:

- "multexp"
    - List / Vector (between 1 and 8 items)
    - "shutter" (only one of the options, 100% of the current shutter if not populated)
        - "percent" *[int, 100]*: Percentage of the current shutter;
        - "value" *[int]*: Shutter in microseconds;
    - "gain" (only one of the options, 100% of the current gain if unpopulated)
        - "percent" *[int, 100]*: Percentage of current gain;
        - "value" *[float]*: Gain in decibels;
    - "flash"
        - "1" *[int, 0]*: Percentage of flash at output 1;
        - "2" *[int, 0]*: Percentage of flash at output 2;
        - "3" *[int, 0]*: Percentage of flash at output 3;
        - "4" *[int, 0]*: Percentage of flash at output 4.

The "shutter" and "gain" fields expect only one of the available options. If both are filled in, the percentage will always be used. In addition, for the flash to work properly it is necessary to pre-configure the outputs used as flash.

The request response contains only the "rid" (64-bit integer field), which is a unique identifier value of the request, serving to correlate requests with the resulting images/processing/metadata.

GET_LASTFRAME

This operation uses the latest Preview exposure to generate a JPEG. The only input parameter is:

- "quality" *[int, 80]*: JPEG conversion quality

The response of this command fills the body entirely with the converted image or returns NACK if a problem occurs. For continuous use (video or preview) it is recommended to use the JPEG_PREVIEW event to generate the images.

EVT_GPIO

This event is sent when there is any change in the state of the "dry" inputs (GPIO) of the equipment.

The data sent are:

- "framecount" *[uint64]*: Frame that the event occurred, useful to correlate with exposures;
- "rising" *[int]*: Combination of all inputs where the logical level was 0 to 1;
- "falling" *[int]*: Combination of all inputs where the logical level was 1 to 0;
- "state" *[int]*: Final state of inputs.

To shorten the messages, the data of all inputs are placed in the same variable of integer type, setting the bit with the corresponding offset. For example, if input 3 changes state from 0 to 1 and input 1 has a high level, the corresponding event (in plain mode) would be:

```
{"framecount":1234, "rising":4, "falling":0, "state":5}
```

Pois, setando o bit correspondente à entrada 3 seria equivalente à `1<<(3-1) => 4` ("<<" sendo operação de shift-left) e a combinação dos bits 1 e 3 seria `(1<<(3-1))|(1<<(1-1)) => 5` ("|" sendo a operação OR bit-a-bit).

For, setting the bit corresponding to input 3 would be equivalent to `1<<(3-1) => 4` ("<<" being a shift-left operation) and the combination of bits 1 and 3 would be `(1<<(3-1))|(1<<(1-1)) => 5` ("|" being the OR bit-a-bit operation).

SET_SERIAL_CFGS

SET_SERIAL_CFGS configures the serial interfaces. They can be used by more than one client, reconfiguration using different parameters, can lead to data loss. It is recommended (valid mainly for RS485 connection) that all equipment on the same bus use the same settings. The configurations available are:

- "serial1"
  - "baud" *[int, 115200]*: Bit rate per second (baud) of the interface;
  - "bits" *[int, 8]*: Number of bits in each frame. Valid values are between 5 and 8;
  - "stop" *[int, 1]*: Number of stop-bits. Valid values are 1 or 2;
  - "parity" *[string, "n"]*: Parity of each frame. Valid values:
    - "n": No generation or checking;
    - "o": Odd parity (odd);
    - "e": Even parity.
- "serial2"
  - Same parameters as "serial1".

EVT_SERIAL

Event generated when data is received via serial interface. Since there is more than one interface, the message format is Mixed. In the metadata, it is sent:

- "pipe" *[string]*: Data source ("serial1" or "serial2")

In the rest of the message (binary format) the received data is sent. In each message, between 1 and 2048 bytes of raw data can be sent. Due to internal timings and other factors, messages being received may be sent in fragments smaller than the maximum size.

SEND_SERIAL_DATA

This operation sends data to the specified interface. The required fields are:

- "pipe" *[string]*: Target interface ("serial1" or "serial2");
- "ascii" *[string]*: Data in ASCII format (no special characters/bytes) or;
- "base64" *[string]*: Data in base64 or format;
- "hex" *[string]*: Data in hexadecimal format (2 characters per byte, 0-9;A-F).

Only one of the data fields is used (using primarily "ascii", then "base64" and then "hex" if multiple are sent). It is recommended to send only the most convenient format.

If everything happens as expected, the return of the command will contain:

- "len" *[int]*: Number of frames (usually bytes) sent.

SET_EQUIP_CFGS

SET_EQUIP_CFGS is the main interface for controlling equipment settings in Cougar. To get closer to the formatting of the REST interface, it controls the image taking settings, equivalent to the REST endpoints:

- /camera/misc/readonly/volatile → "equip.volatile";
- /camera/misc/readonly/constants → "equip.miscRO";
- /camera/misc → "equip.misc";
- /camera/autofocus → "equip.autofocus";
- /camera/profiles/:id (except profile creation and deletion) → "equip.profiles";
- /camera/profiles → "equip.currProfile";
- /camera/transitioner → "equip.transitioner";
- /camera/ios → "http://equip.io ".

It also controls the Services endpoints:

- /camera/services/stream → "h264" e "mjpeg";
- /camera/services/ocr → "ocr";
- /camera/services/classifier → "classifier";
- /camera/services/analytics → "analytics";
- /camera/services/ftp → "ftp";
- /camera/services/itscampro → "itscampro";
- /camera/services/licenses → "license".

The input parameters (optional) for this command are:

- "path" *[string,""]*: Root path to consider for the data;
- "date" *[object/value, {}]*: Data to be written in settings.

All previously mentioned settings are available in a full setting tree. Therefore, like all other Cougar settings, it is possible to configure the elements sparsely, configuring only what is necessary. This makes, from the point of view of writing, the following options equivalent:

1) {"path":"analytics.voting.roi1.x0", "data":123}
2) {"path":"", "data":{"analytics":{"voting":{"roi1":{"x0":123}}}}}
3) {"path":"analytics.voting", "data":{"roi1":{"x0":123}}}

Due to the total volume of settings, when using empty "path", in a conventional case, around 800 elements are answered. This unnecessarily increases the consumption of processing and network resources and is therefore not recommended for continuous use. For the examples above, the answers are:

1) 123;
2) all settings (~800 elements);
3) majority voting settings only (~25 elements).

Due to the volume of settings, it is recommended to use the rest interface documentation as a reference to the parameters of this command.

Available APIs

Once the protocol is opened, it is possible to implement its own version of the client, especially if its application does not use one of the languages in which they were implemented, or if the version made available does not meet the requirements of the rest of the application (for example, python 2 vs. 3, C++ 98 vs. 11 vs. 17 vs. 20). Pumatronix makes customers available in C++ and Python, for code reference only (please contact technical support to get them).

General Recommendations

The settings can be changed partially/sparsely, without the need to read, modify and write a fixed structure. In this way, even if new fields are added, customer behavior remains consistent.

For the SET_EQUIP_CFGS operation, one should try to inform the path closest to the settings that are desired to be set, making the return of the operation as small as possible. It is recommended to obtain all settings ("path": "") only when necessary (for example: when initializing customer logic).

To implement the protocol independently, it is highly indicated to use a multi-threaded environment to handle the reception and interpretation of messages on a separate thread, to reduce latency between data reception. If necessary, it is possible to use a *threadpool* so that interpretation is done faster in multi-core environments.

All messages are asynchronous, that is, during the processing of a command (waiting between sending and receiving) other messages (events, images or even commands from other threads) can be received.

XMODEM CRC16 Calculation Example

```
// Example of implementation of the CRC16 XMODEM algorithm. This example uses a
// CRC tab" to precalculate coefficients and optimize performance.
// You can define the tab as a constant of "magic" values.
// It uses defined variables in <stdint.h> which they can be replaced
// by the equivalent of the system. They usually are:
// - uint8_t -> unsigned char
// - uint16_t -> unsigned short int
uint16_t crcXMODEM(const uint8_t *c, int numBytes, uint16_t currCrc = 0) {
    static uint16_t crcTab[256];
    static bool init = 0;
    if(!init){
        init = true;
```

```
        for (int i=0; i<256; i++) {
            uint16_t short_c = i<<8;
            uint16_t crc = 0;
            for (int j=0; j<8; j++) {
                if ( ((crc ^ short_c) & 0x8000) != 0 ) {
                    crc <<= 1;
                    crc ^= 0x1021;
                } else {
                    crc <<= 1;
                }
                short_c <<= 1;
            }
            crcTab[i] = crc;
        }
    }
    for (int i = 0; i < numBytes; i++) {
        currCrc = (currCrc << 8) ^ crcTab[((currCrc >> 8) ^ c[i])];
    }
    return currCrc;
}
```

4. Pumatronix Open-Source Communication Protocol (Socket)

Communication with the ITSCAM 600 is done through the Ethernet interface, using the Pumatronix UDP and TCP/IP Communication Protocol. The port used for communication with external equipment is number **50000**. Thus, the application developed to communicate with the ITSCAM 600 must be configured to send commands using this port in the TCP and UDP protocols.

The UDP protocol is used only for the identification of equipment connected to the network, as it allows the sending of *broadcast* packets, which are received by all devices. This allows ITSCAM 600 to send their identification upon receiving this package. All other commands use the TCP protocol, which establishes a point-to-point connection between the control device and the ITSCAM 600. Security in receiving and changing settings occurs with the transmission of a *CRC* code. However, most of the responses sent by the equipment do not have *CRC*.

The Pumatronix Protocol supports connections that have been developed based on the Dynamic Library (dll) and the C++ class for Linux. At www.pumatronix.com a *Development Kit (SDK)* is available with the files necessary for the development of the application.

To download it, go to the *Customer Area > Technical Support.*

Commands
VIDEO
VIDEO_CONTINUO

FOTO
FOTO_INFO_ADIC
REQ_CONF_IMAGEM
REQ_CONF_GER
SETA_DEFAULT
REQ_CONF_REDE
REQ_SITUACAO_DAY_NIGHT
REQ_ROTACAO
REQ_LENTE_AUTO_IRIS
REQ_MODO_DAY_NIGHT
REQ_SATURACAO
REQ_WHITE_BALANCE
REQ_FOTO_COLORIDA
REQ_HORA
REQ_DATA
REQ_ENTRADA
REQ_GAMMA
REQ_GANHO_VISIVEL
REQ_TEMPO_LIGADO
REQ_MOV_LENTE
REQ_FOCO_IR
REQ_IO_VIGIA
REQ_DIR_IO_VIGIA
SETA_ROTACAO
SETA_LENTE_AUTO_IRIS
SETA_MODO_DAY_NIGHT
SETA_SATURACAO
SETA_WHITE_BALANCE
SETA_FOTO_COLORIDA
SETA_SAIDA
SETA_TIPO_SHUT
SETA_SHUT
SETA_SHUT_MAX
SETA_TIPO_GANHO

SETA_GANHO
SETA_GANHO_MAX
SETA_NIVEL_IMG
SETA_ZOOM
SETA_FOCO
SETA_POSICAO_ZOOM
SETA_POSICAO_FOCO
SETA_IO_VIGIA
SETA_DIR_IO_VIGIA
SETA_GAMMA
REQ_WB_ATUAL
SETA_ZOOM_FOCO
REQ_POSICAO_ZOOM
REQ_POSICAO_FOCO