

# **TVObjects gadget manual**

## Table of Contents

1.	What is TVObjects gadget? .....	3
1.1.	Input and output pins.....	3
1.2.	To which gadgets can TVObjects gadget connect? .....	4
1.3.	TVObjects gadget description.....	5
1.4.	What is it Object?.....	6
1.5.	What is it Task?.....	6
1.6.	Parameters of object.....	7
1.6.1.	Common controls for all objects .....	7
1.6.2.	Common parameters for all objects.....	8
1.6.3.	Parameters for spot object .....	11
1.6.4.	Parameters for line object.....	18
1.6.5.	Parameters for edge object .....	20
1.6.6.	Parameters for text object.....	24
1.7.	Data flow and formats.....	26
1.8.	Gadget properties. ....	32
1.9.	New video object creation.....	32
1.10.	Control of objects parameters by text frames.....	34
1.11.	Additional possibilities for TVObjects gadget control by text frames.....	34
1.11.1.	Scale control .....	35
1.11.2.	Task switch .....	35
1.11.3.	View mode switch .....	35

# 1. What is TVObjects gadget?

TVObjects gadget determines which objects will be searched for in the received video data frames and sends the information regarding the detected objects to output pin.



Figure 1: TVObjects gadget

## 1.1. Input and output pins

Pin 1 (upper left side of the gadget) is used to receive the video data frames and/or gadget or objects properties. Video frame could be inside of container frame(s). The first found video frame will be used for processing. Keep in mind that it will be last attached to container video frame (searching is going from last to first attached).

Pin 2 is used to receive task execution commands (text string such as **"Task(0);"** causes the gadget to search for the objects defined in Task(0) in the received video frames till further notice).

Pin 3 is used to receive objects-to-be-searched descriptions (objects and tasks could be alternated through this pin).

Pin 4 is used to receive coordinates for measurements or commands for measurement areas changing. Usually, this pin is used for interactive process control.

All commands will be received as text (by text frames).

The output pin 5 is used to send the information regarding the detected object along with the original video data frame.

The output pin 6 is used for diagnostics information providing (process time, time from image capturing and so on).

## 1.2. To which gadgets can TVObjects gadget connect?

Since the first input pin is used to receive video frames, it can connect to any video output pin (such as PtGrey1\_8 camera gadget or AviCapture gadget).

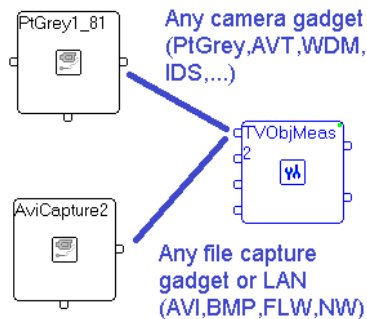


Figure 2. Input pin 1 connection

Pin 1 has **a queue for 4 input frames** and one processing thread, which takes input frames for processing one by one. If the queue is full, arrived input frames **will be rejected** (thrown).

The rest input pins receive text strings and can therefore connect to any text output pin (such as on GenericCapture, TextCaptureCtrl, NW – network...).

Pin 2 could be used for dynamic processing control (about tasks selection look below in part 2)

Pin 3 could be used for visual objects description change (also look below in part 2).

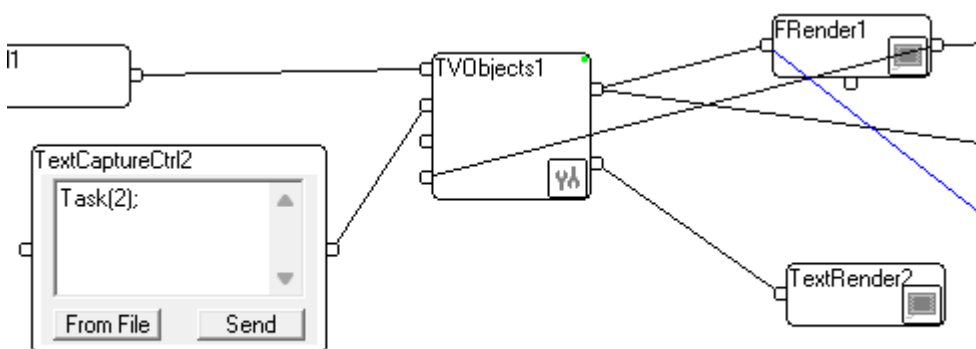


Figure 3. Task control by input pin 2.

The pin 4 receives text strings containing coordinates for auto detection, object selection or ROI movement and therefore should be connected to any text output pin that sends coordinates to its output pin. Video renderer does such operation (look on Figure 4; connection line from FRender output pin and pin4 of TVObjects).

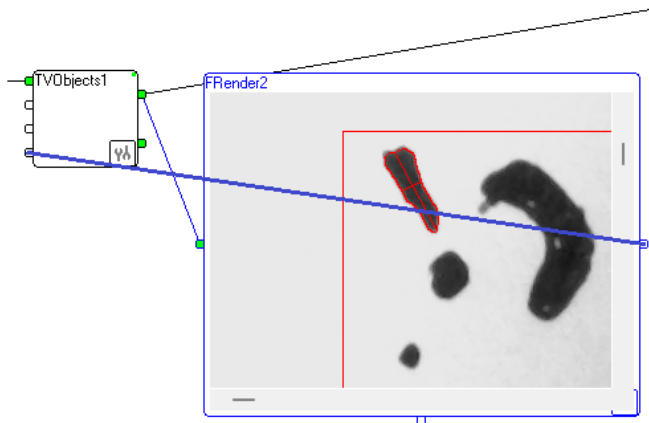


Figure 4. Input pin 4 connection

The information regarding the detected objects is sent to the output pin in a predetermined data structure which can be deciphered by some gadgets (FieldChooser, ScriptRunner) or by host application. If the output pin is connected to a video render gadget, this render will show the original video stream with overlay of the detected object parameters and the ROI (Region of Interest).

### 1.3. TVObjects gadget description

Double left click on TVObjects gadget creates the following dialog box:

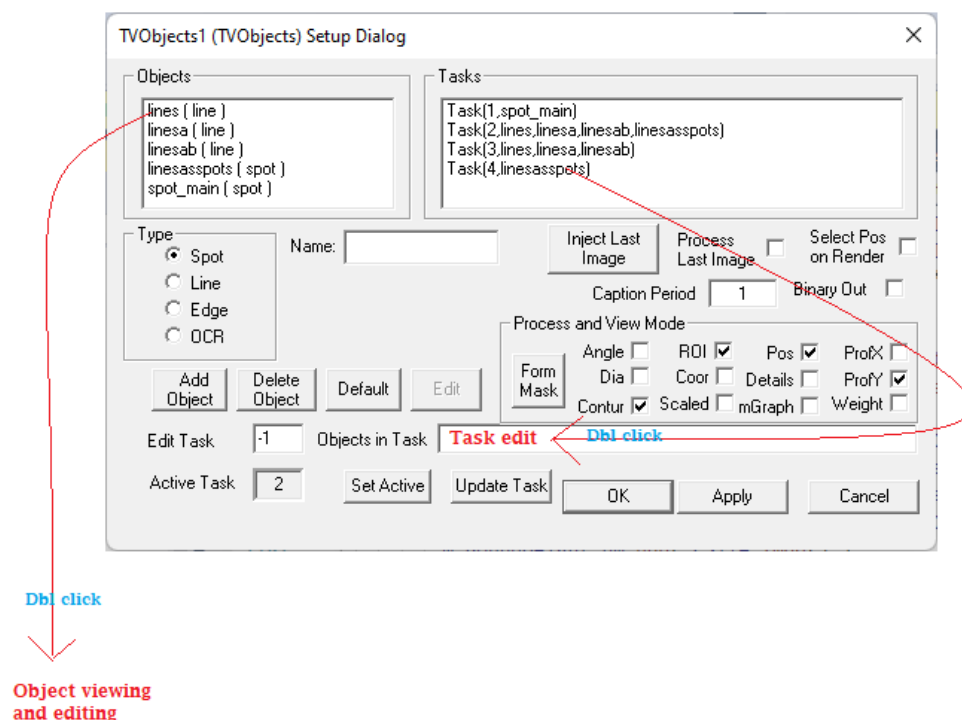


Figure 5. TVObjects Setup Dialog

The upper part of this dialog consist of two list boxes: Objects list on left side and Task list on right side.

Double click by left button on Object in left list box leads to viewing of additional dialog for object parameters viewing and editing.

Double click by left button on Task in right list box leads to selected task moving into task edit box.

## 1.4. What is it Object?

Object is some measurable picture element. TVObjects gadget knows 4 kinds of such elements: spot, line, edge and text. Description of these elements as set of parameters for searching, measurement, filtration and viewing you can find below. Sometime object will be called '**video object**'.

## 1.5. What is it Task?

The task is list of objects, which should be found, measured, filtered and viewed. When image arrived to pin 1 of TVObjects gadget the selected (Active) Task will be performed for task objects detection and measurement.

Task description is one string consisting of items separated by comma. The first element is task number which is used for task identification. The rest of this string is object names for selection from object shown in left list box of dialog. Objects will be found and measured in order **from left to right** in the task objects list.

Task could be edited manually in edit box "Objects in Task". If some name in task doesn't correspond to name in object list, this object will be not measured.

## 1.6. Parameters of object

### 1.6.1. Common controls for all objects

Double click on object in left list box (Figure 5) lead to object editing dialog viewing. Form of this dialog is dependent on object type and on object placement parameter.

Upper part of edit dialog for all objects has following view:

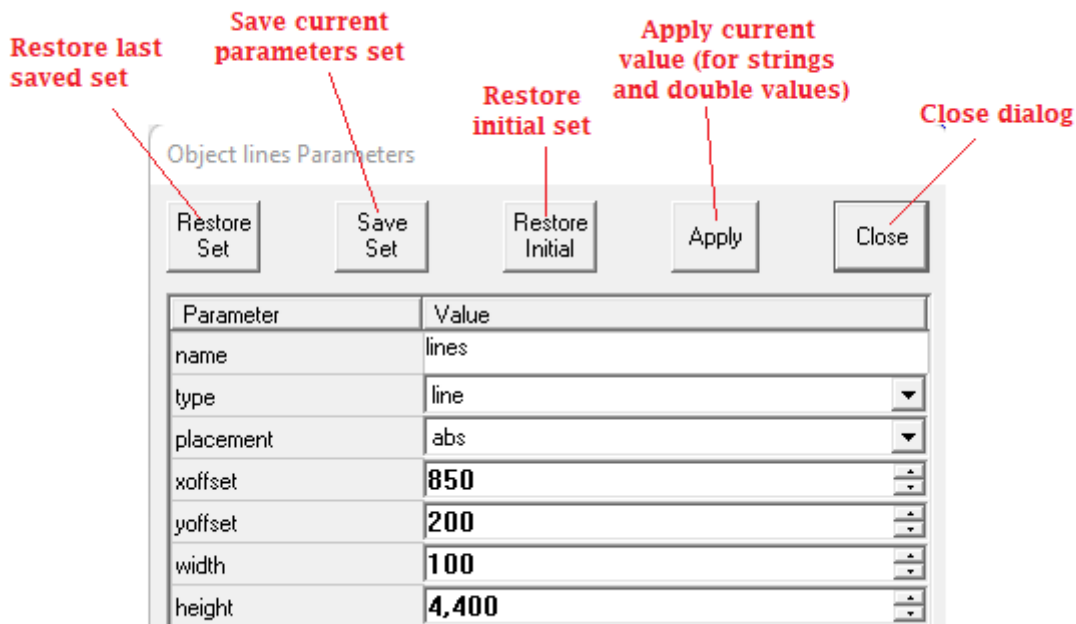


Figure 6. Upper part of setup dialog for video object.

Dialog has several buttons:

“Save set” – current parameters set will be saved in gadget. When gadget will be deleted from graph, this saved set will be lost.

“Restore Set” – dialog will be reinitialized with saved set. Keep in the mind, current version has only one saved set.

“Restore Initial” – will be restored parameters set which was used when graph was loaded.

“Apply” – if string or double parameter was edited, press on this button will apply value to video object. The same will be done, when user begin to use other than edited dialog item.

“Close” – will close dialog. Pressing on ESCAPE keyboard key also will close dialog.

### 1.6.2. Common parameters for all objects

"Name" – object name edit box(string). Any letters and digits could be used. Digit in first position is allowed. Keep in the mind, that there is no difference between capital and small letters (case insensitive)

"Type" – object type should be selected in this list box. Spot, line, edge or text are known values.

"Placement" – object placement list box. There are 3 known placement types:

- Abs – search zone for object has absolute position (dialog for this case shown above). XOffset and YOffset values are relatively to image origin (point[0,0]).
- Rel – search zone is relative to **found** position of another object. Dialog fragment view for this case is below. Additional item with name of "anchor" object ("spot\_main") is shown below. XOffset and YOffset values are relatively to "anchor" object

type	line
placement	rel
relativeto	spot_main
xoffset	480
yoffset	-140
width	100

- RelXY - search zone is relative to **found** positions of two another objects: XOffset is relatively **found** X position of first ("spot\_main") object, YOffset is relatively to Y coordinate of **found** second ("linesab") object.

type	line
placement	rel_xy
relativeto	spot_main
relativetoy	linesab
xoffset	-260
yoffset	-580
width	100
height	4,400

Keep in the mind, that anchor objects should be measured before current object, i.e. in task description string their names should be on the left side from current object.

"XOffset" and "YOffset" – coordinate of left-top corner of object search area.

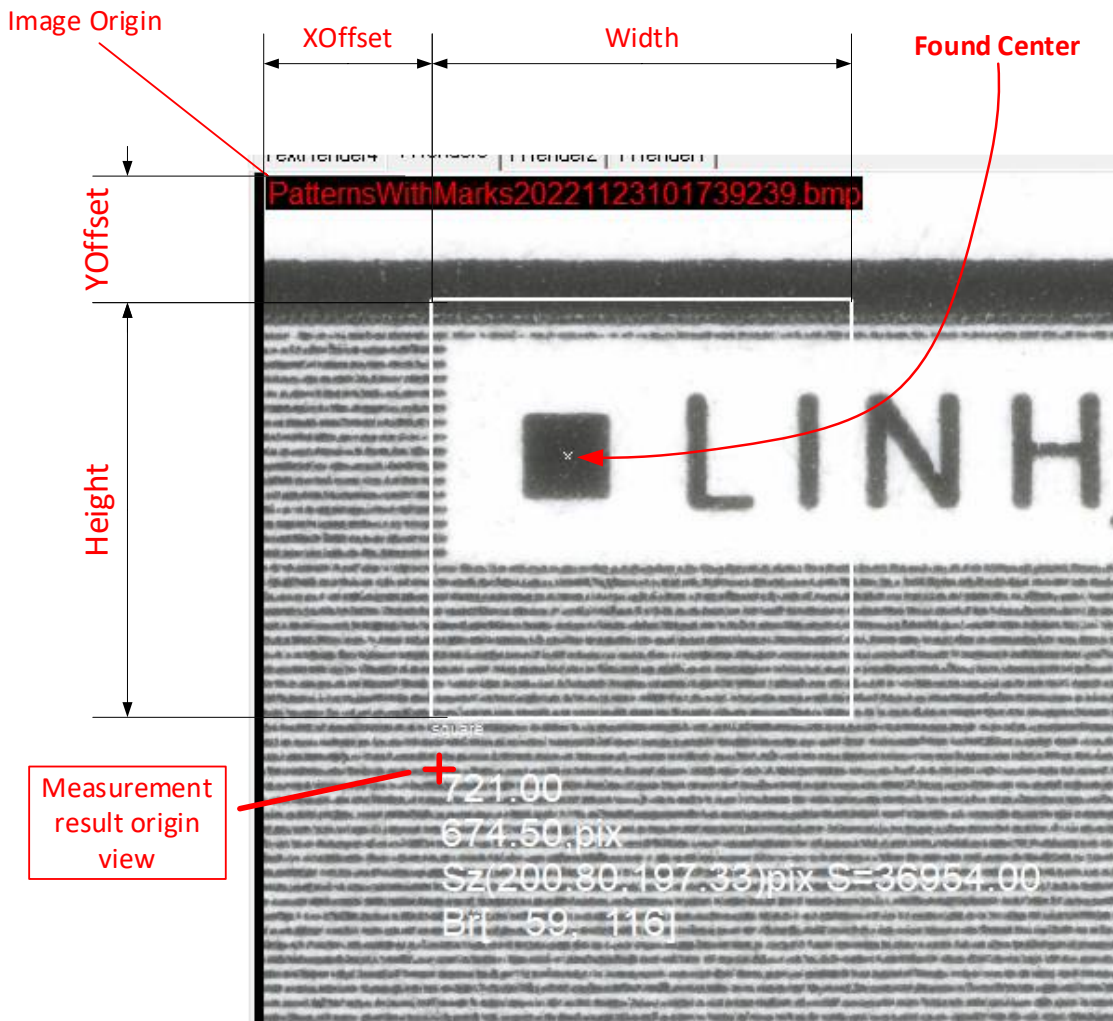
"Width" and "Height" – width and height of object search area

Below are fragment of dialog with search area parameters and illustrating image.

The object is black square with some parameters. White square is search area. Black square in this search area should be found and measured.



name	square
type	spot
placement	abs
xoffset	400
yoffset	300
width	1,000
height	1,000

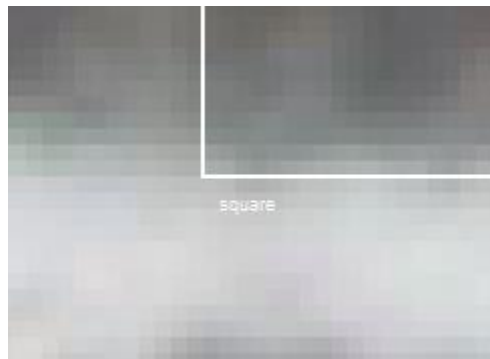


Following parameters are for result viewing.

color	white
textsize	16
xtextoffset	-300
ytextoffset	740

“Color” – list box for result view show (per object). The same color will be used for result text and for search area marking (white square above). If object is not found, the searching area will be shown as “negative” color (for example, red will be shown as cyan, i.e. green

and blue). If search area is shown the name of object will be written near bottom left corner (it's visible on image above or on zoomed fragment below).



“Textsize” – text size for results viewing. Bigger number lead to bigger text.

“XTextOffset” and “YTextOffset” – offset of result viewing point relatively to found object center (for text recognition this position is relatively to search area left-top corner).

There are groups of check boxes for measurements and viewing control. Purpose of these check boxes will be explained in following spot parameters description section.

### 1.6.3. Parameters for spot object

Spot is an area, differing in brightness from the surface around it. TVObjects gadgets deals with light intensity only. As result, only grey scale images (formats Y8, Y800, Y16) and color planar formats (YUV9 and YUV12) will be processed. Other formats should be converted to one of these.

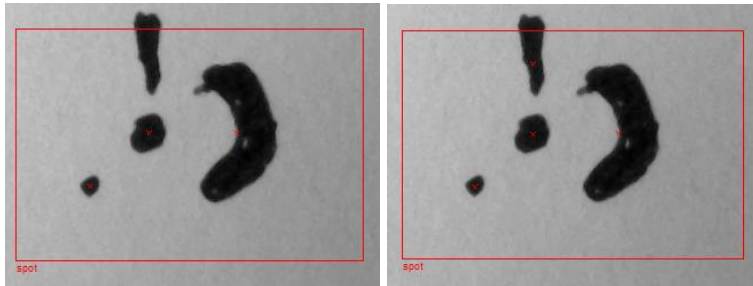
All black objects on following image are spots.



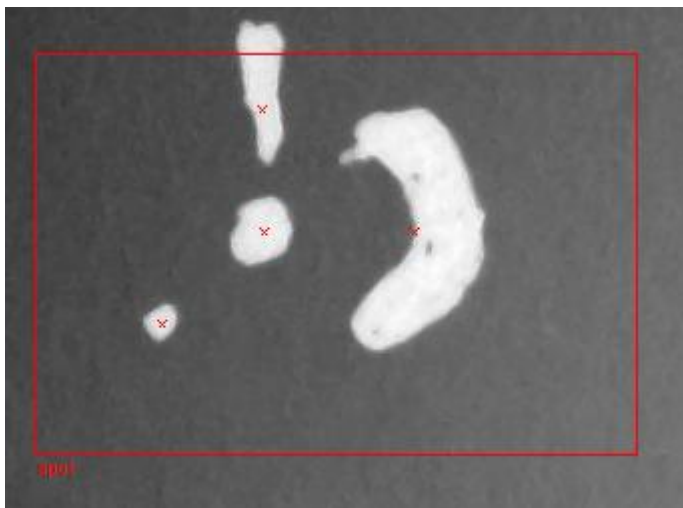
Fragment of Object setup dialog with spot parameters is shown below.

<input type="checkbox"/> <b>donttouchedge</b>	
contrast	white_on_black
width_min	5
height_min	5
width_max	100
height_max	150
area_min	15
area_max	10000
xdiffr	0
ydiffr	0
<input checked="" type="checkbox"/> <b>multi</b>	
timeout_ms	0
max_n_objects	85
min_ampl	20
thres	0.500
rotation_thres	0.500

“DontTouchEdge” check box: if it’s checked, spot should not touch edge of search area. Following two images are showing this check box influence: on right one it is checked, on left is not (look on small red cross on center of upper spot which crosses search area border: on left image it’s not measured, on right the cross is on the center of this spot part which is inside of the search area).



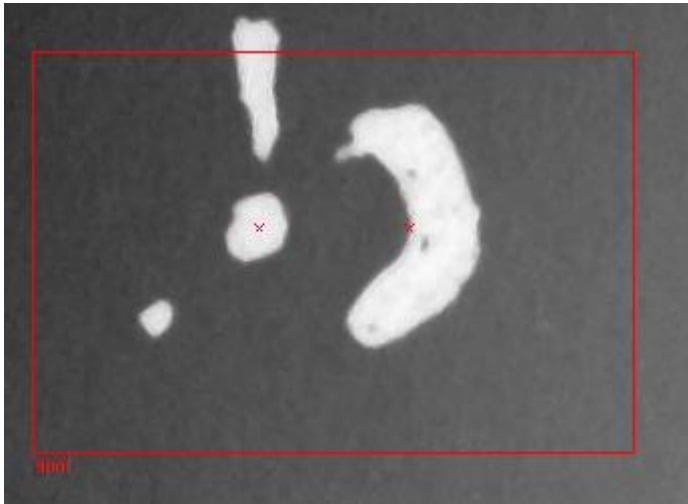
“Contrast” – spot contrast relatively to surrounding. Could be “black-on-white”, “white-on-black” or “any”. Previous image was measured with “black-on-white” contrast. The same result will be obtained with “any”. Following image is negative and could be measured with “white-on-black” and “any” selection as contrast (and measured with unchecked “DontTouchEdge”).



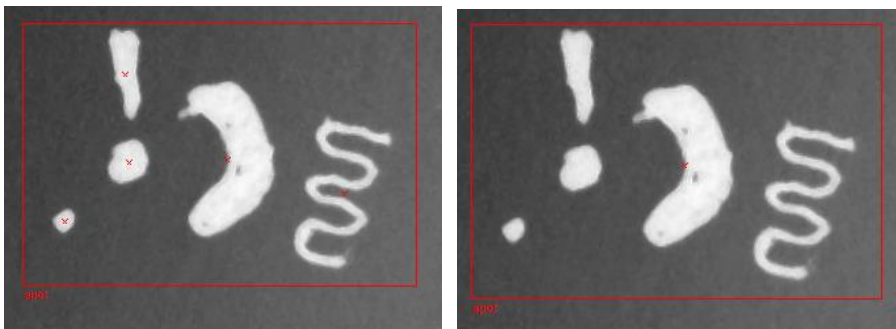
“Width\_min”, “Width\_max”, “Height\_min” and “Height\_max” – this 4 parameters are defining min and max object sizes in pixels. Spots on previous 3 images are measured with following values

width_min	5	
height_min	5	
width_max	100	
height_max	150	

If “Width\_min” will be changed to 30, upper and left spots will be filtered out (widths are too small).

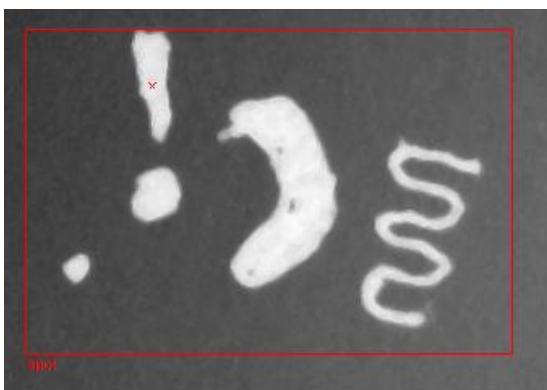


“Area\_min” and “Area\_max” – values of min and max of area in square pixels for objects filtration. Following left image is after processing with area range [50,5000], the right image is for area range [3000,5000] – on right image only “big” object near center is marked as measured.



Parameters “xdiffr” and “ydiffr” are for laser or led beam measurements.

“Multu” check box is for multi objects measurement. All previous measurement done with it’s checked. If it’s unchecked following measurement result will be obtained: only one object is measured – program does image scanning from left to right and from up to down. After first measured object process was stopped.



“timeout\_ms” – this is approximate object processing maximal time in milli seconds. Processing algorithm sometimes does elapsed time checking and if this time more than appointed processing will be aborted. If this value is zero then no time restrictions applied.

“Max\_n\_objects” – maximal number of measured objects. If processing algorithm detects more objects, than appointed processing will be aborted.

“Min\_ampl” – minimal image contrast (difference between min and max brightness). If image contrast in search area is less than this value, processing will be aborted.

“Thres” – processing threshold. If this value is in range (0.,1.) then threshold is relative to minimal and maximal brightness in search area.

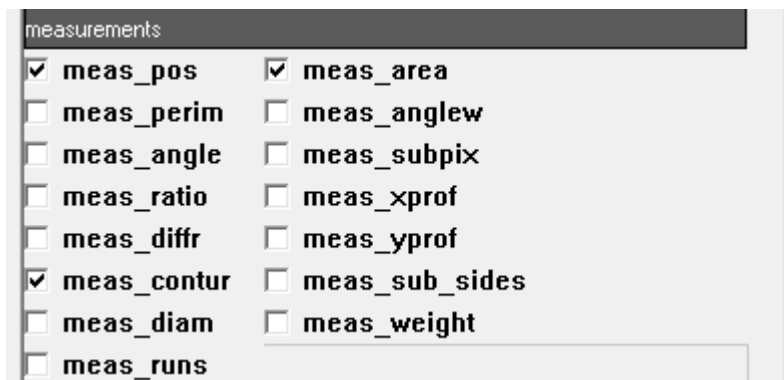
$$\text{Threshold} = I_{\min} + \text{Thres} * (I_{\max} - I_{\min}),$$

where  $I_{\max}$  and  $I_{\min}$  are maximal and minimal intensities in search area.

If this value more than 1.0, value will be used as absolute value. Keep in the mind, that usually images have 8 bits per pixel, i.e. threshold should be in range from 1 to 254. If image with format Y16 is used, threshold could be in range 1 to 65534.

“Rotation\_thres” is used for laser or LED beam measurement.

Measurement check boxes section for spot processing control are shown below.



Measurement results could be provided with pixel accuracy or subpixel accuracy. If “meas\_subpix”, “meas\_contur” or “meas\_weight” are checked the output data will be with subpixel accuracy, other wise accuracy will be 0.5 or 1.0 pixel.

“Meas\_pos” check box controls object position measurement. When position is measured the object center coordinates will be saved in output parameters.

“Meas\_area” controls object area measurement. When area is measured the object center coordinates will be saved in output parameters.

“Meas\_perim” controls perimeter measurement. Result will be in pixels.

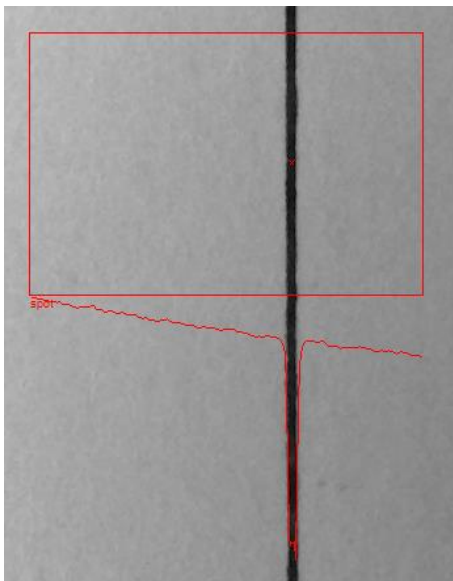
“Meas\_angle” controls measurement of binarized spot image long diameter direction with each pixel weight as 1. Result will be in degrees in range  $[-\pi/2, \pi/2]$ .

“Meas\_anglew” controls measurement of binarized spot image long diameter direction with each pixel weight as pixel intensity. Result will be in degrees in range  $[-\pi/2, \pi/2]$ .

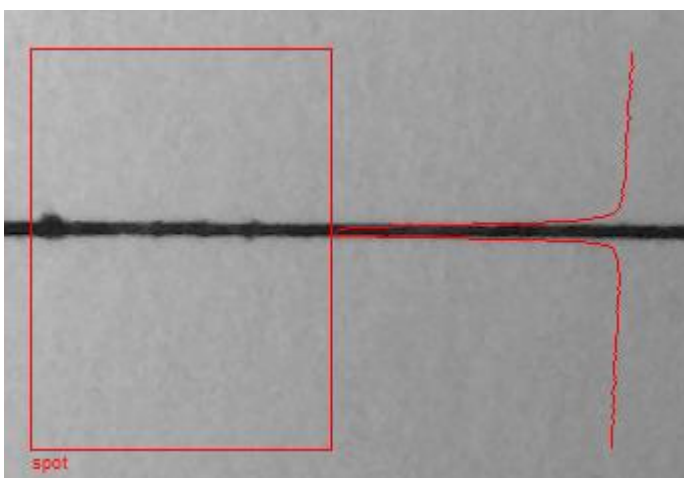
“Meas\_subpix” controls subpixel measurements for position.

“Meas\_ratio” controls measurement of ratio between binarized image long diameter and orthogonal direction on binarized image (look “meas\_dia”)

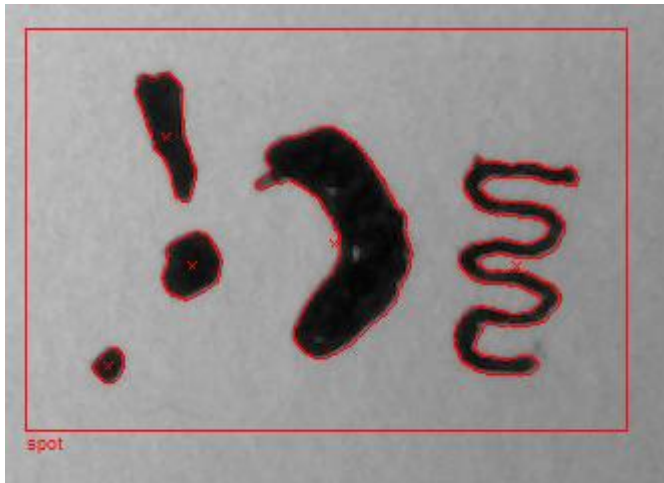
“Meas\_xprof” controls X profile measured: sums of all pixels in each column in search area will be divided on search area height and saved as array and sent to output. Following picture show measured X profile. Viewing is normalized – (max-min) is scaled to search area height. View will be placed below of the search area, but if below is not enough place, the profile view will be placed above.



“Meas\_yprof” controls Y profile measured: sums of all pixels in each row in search area will be divided on search area width and saved as array and sent to output. Following picture show measured Y profile. Viewing is normalized – (max-min) is scaled to search area width. View will be placed bon the right side of the search area, but if on the right is not enough space, the profile view will be placed on the left side.

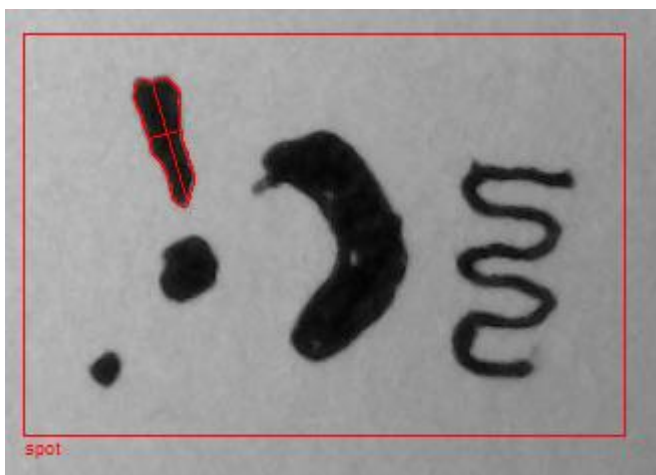


“Meas\_contur” controls contour around spot. Sample of contour measurement (and viewing) shown on picture below.



“meas\_sub\_sides” control is used for laser or LED beam measurements.

“Meas\_diam” controls diameters measurement. As result two vectors for long and short diameters will be sent to output. Look following picture with diameters presentation for one spot filtered by area.

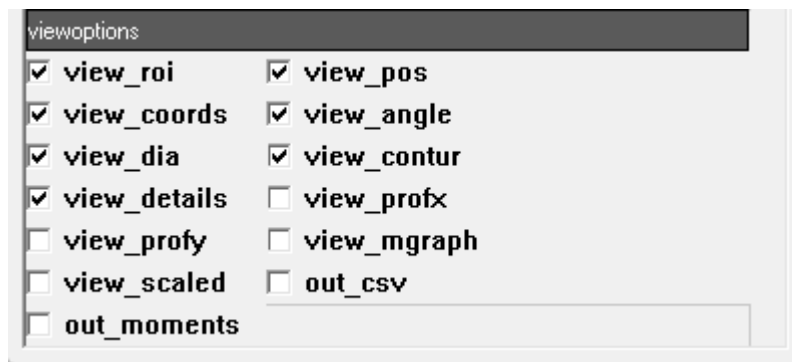


“Meas\_weight” controls spot center measurement: if it is checked the weighted center of gravity measurement will be done, i.e. each pixel intensity will be used.

“Meas\_runs” controls spot runs saving, i.e. save segments of overlap between spot and each image row.



Viewing check boxes section for spot data presentation control are shown below.



“View\_ROI” controls search area show.

“View\_pos” control show of cross on spot center position.

“View\_coords” controls short presentation of data about spot (center position, spot height and width, perimeter, area and brightness range).

“View\_angle” controls spot angle viewing if “View\_coords” is ON.

“View\_dia” controls spot diameters showing on image (as two line segments crossing in center).

“View\_contur” controls contour around spot viewing if contour is measured.

“View\_details” and “View\_mgraph” are used for laser and LED beams measurement results viewing.

“View\_profx” controls profile X viewing (if it’s measured).

“View\_profy” controls profile Y viewing (if it’s measured).

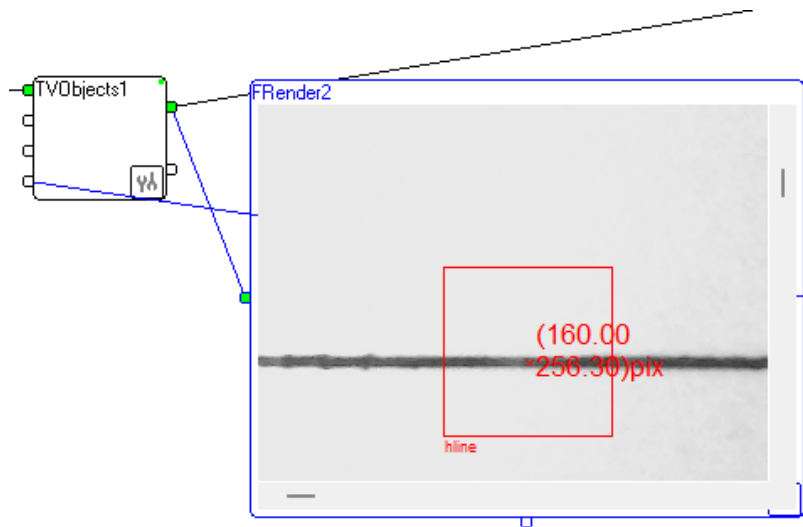
“View\_scaled” controls view with coordinates and sizes conversion from pixels to other units used in gadget (look below about scale and units setting into gadget).

“out\_csv” controls data output in CSV format: in first versions of SH the output data about spot was a string with as separation symbol. If this check box is settled, data will be provided with comma separation. This provided for direct export to Excel file.

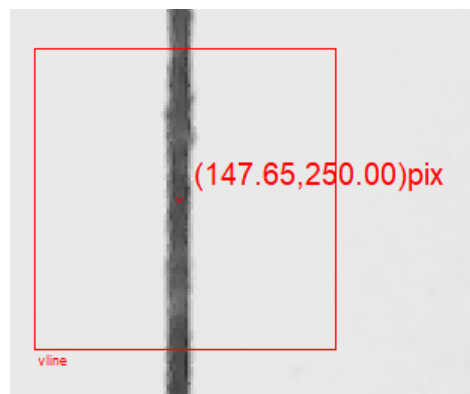
“Out\_moments” – if this control is settled special string with image moments will sent to output. About image moments look [https://en.wikipedia.org/wiki/Image\\_moment](https://en.wikipedia.org/wiki/Image_moment).

#### 1.6.4. Parameters for line object

Line in SH is video object crossing search area with relatively big ratio between width and height. TVObject deals with horizontal and vertical lines. Following image shows sample of horizontal line.



Sample of vertical line is below.



Setup dialog fragment for line object shown below.

Parameter	Value
name	vline
type	line
placement	abs
xoffset	100
yoffset	200
width	100
height	100
dir	vertical
contrast	black_on_white
thick_min	1
thick_max	10
<input type="checkbox"/> multi	

Setup for line object has 3 different from spot object parameters.

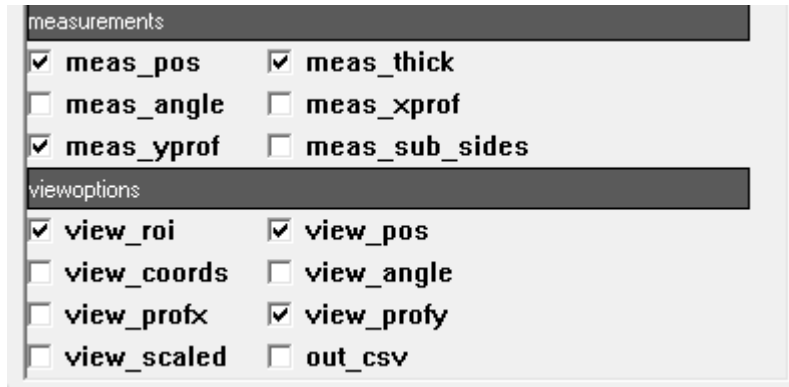
“Dir” controls line orientation, which could vertical or horizontal.

“Thick\_min” is used for objects filtration and holds minimal line thickness.

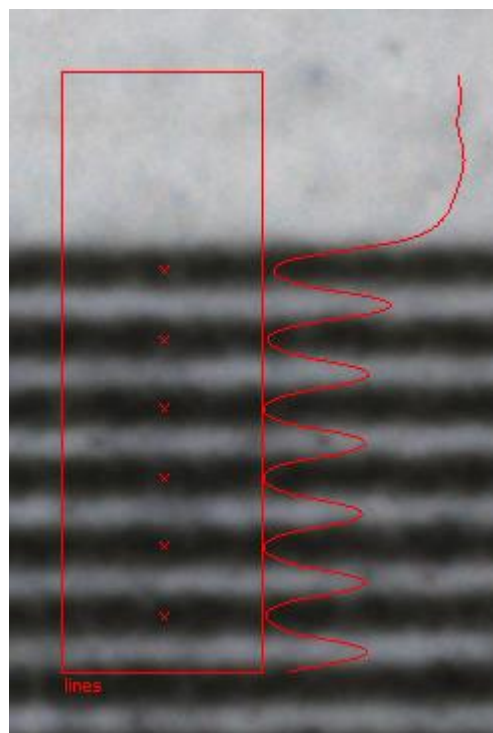
“Thick\_max” is used for objects filtration and holds maximal line thickness.

Only lines with thickness from “thick\_min” to “thick\_max”

There is control check boxes setup dialog fragment for line object.



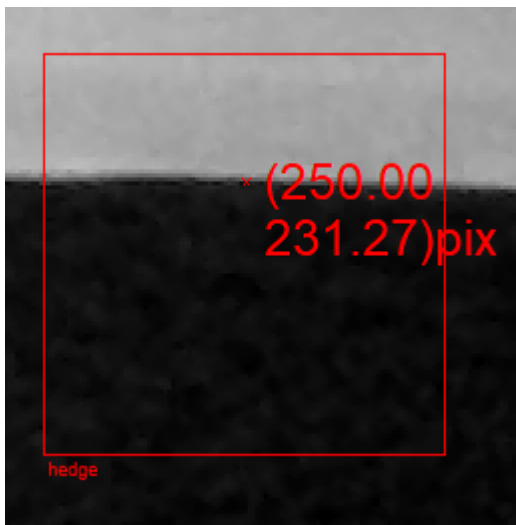
Below is image fragment for line object measurement when “multi” check box is checked and with profile calculation and view.



#### 1.6.5. Parameters for edge object

The edge object is straight border between two image regions with different intensity.

Horizontal edge is shown on following image.



TVObject gadget deals with horizontal and vertical edges only. Other directions could be processed in special mode “edge\_as\_contur”, when raw data about edge will be generated (look in description for check boxes for line). Also other directions could be handled by gadgets after TVObjects with detected edges analysis. There are 3 controllable parameters for edge object

name	edges
type	edge
placement	abs
xoffset	2,202
yoffset	2,202
width	100
height	100
debounce	0
dir	downtoup
contrast	white_to_black

☐ multi

“Contrast” list box has 3 possible selections:

- When white\_to\_black is selected, the transition from brighter area to darker will be detected
- When black\_to\_white is selected, the transition from darker area to brighter will be detected
- When any is selected, the any transition will be detected.

“Dir” controls transition search direction. There are 4 possible selections:

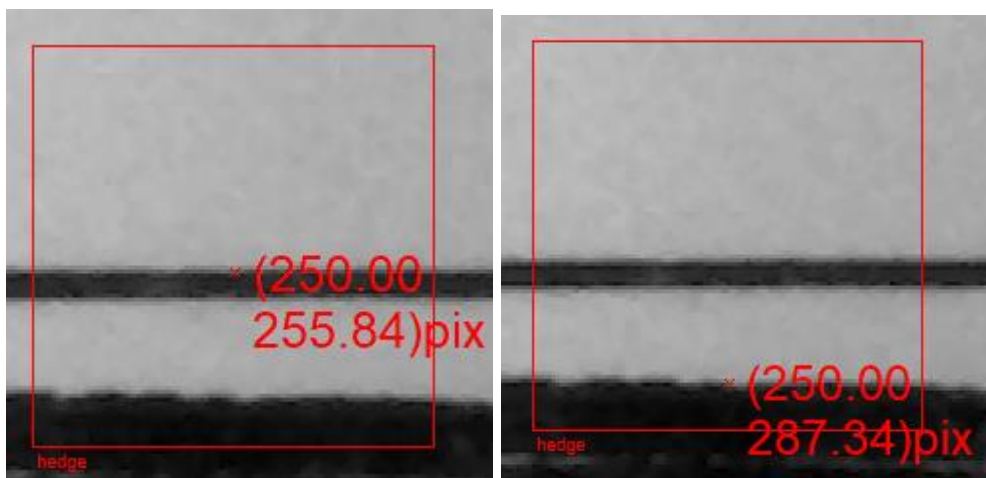
- Lefttoright
- Righttoleft
- Downtotop
- Toptodown

For horizontal edge should be detected the only “downtotop” and “toptodown” should be used.

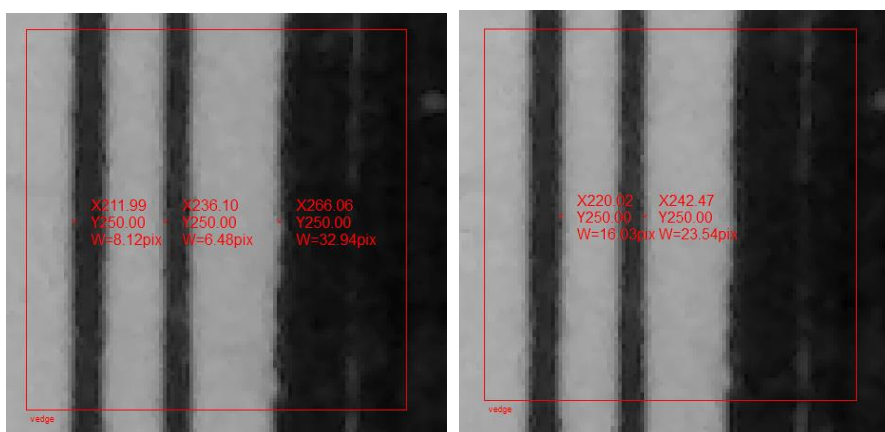
For vertical edge should be detected the only “lefttoright” and “righttoleft” should be used.

“Debounce” is used for too thin lines filter out.

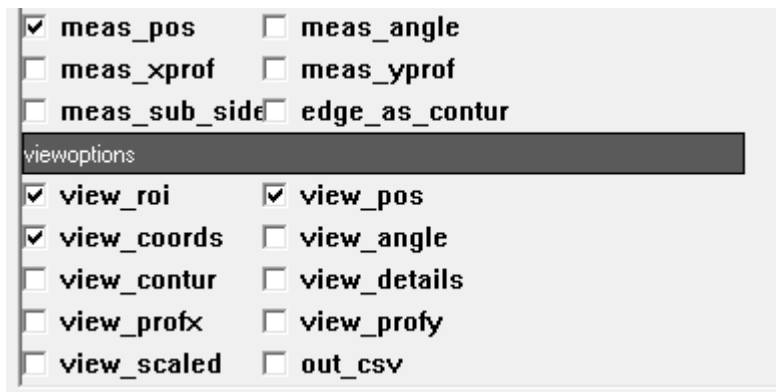
Following picture illustrates debouncing: the left image is result of processing with debouncing 2, the right image is result for debouncing 8 (upper black line is not detected).



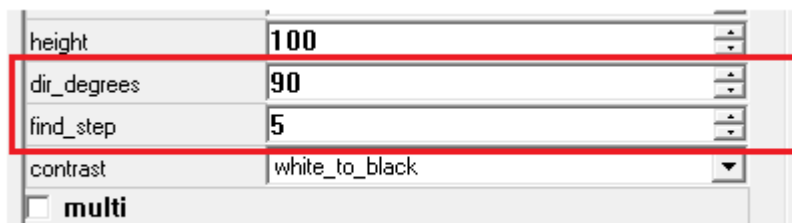
Checkbox “multi” controls multi edge functionality. If it’s checked, algorithm continues to search edges after first one. Following images illustrate this functionality for left-to-right searching for white-to-black edge (left image) and for black-to-white (right image). W value is width of zone after edge until opposite edge direction.



Checkboxes for line measurement and view control is shown below.

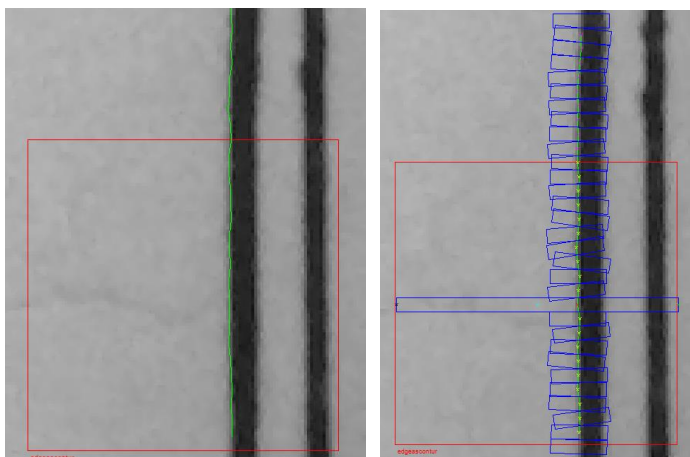


Checkbox “edge\_as\_contur” Specific for edge measurement. If it’s checked the view of setup dialog is changed:



Item “dir\_degrees” defines edge direction: 0 degrees is correspondent from left to right, 90 degrees is from bottom to up. This item can have angles from -360 to +360 degrees. It’s possible use spin of this item in real time, when images to gadget entry coming with rate >5 frames per second, and to see influence.

When “edge\_as\_contur” is checked, the searching process is organized as following: a vector orthogonal to edge direction calculated the angle between edge direction and vector is -90 degrees (vector goes to the edge from the left side). Program calculates averages for “find\_step” pixels around this vector (orthogonal to the left and to the right from vector) and algorithm is looking for average values step (for shown in dialog values for “white\_to\_black”). Average values calculated for ROI zone. The left following image shows found edge by green. If checkbox “show\_details” is checked, possible to see initial search zone (long blue rectangle). Short rectangles do show search areas to the left and to the right from initial found point.



Edge is tracked and marked by green for 1.5 of ROI out of ROI (it shows edge direction as settled in setup dialog; for this example, it's 90 degrees to horizontal direction).

Every blue rectangle, except of initial search, has width "find\_step" and length is twice of width.

Edge could be not straight but should not change direction for more than 45 degrees in ROI.

If edge became too complicated, program aborts processing (edge will be not found).

### 1.6.6. Parameters for text object

Text object is some row of symbols or punctuation marks. TVObjects gadgets deals with light intensity only. As result, only grey scale images (formats Y8, Y800, Y16) and color planar formats (YUV9 and YUV12) will be processed. Other formats should be converted to one of these.

There are several specific for text object parameters (look on following picture):

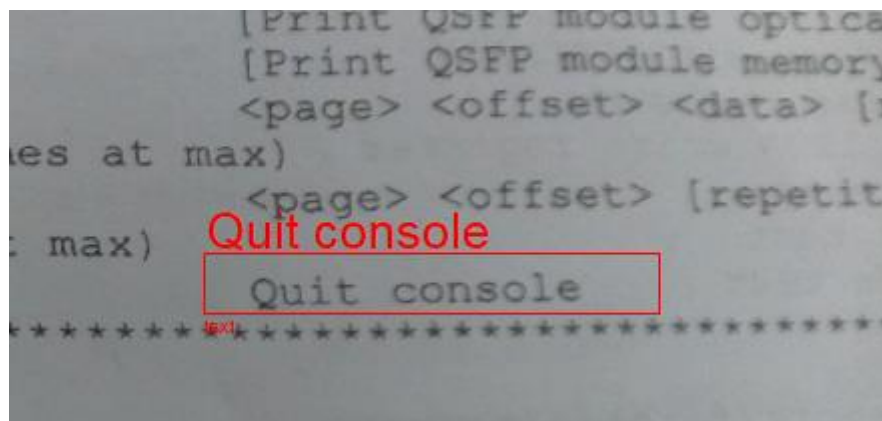
Object text Parameters

Restore Set Save Set Restore Initial Apply Close

Parameter	Value
name	text
type	text
placement	abs
xoffset	'1,147
yoffset	'343
width	'480
height	'64
dir	right_3
timeout_ms	'0
max_n_objects	'10
min_ampl	'20
thres	0.500
color	red
textsize	'20
xtextoffset	'5
ytextoffset	'-54
measurements	
<input checked="" type="checkbox"/> meas_text	<input checked="" type="checkbox"/> meas_tfast
viewoptions	
<input checked="" type="checkbox"/> view_roi	<input checked="" type="checkbox"/> view_text

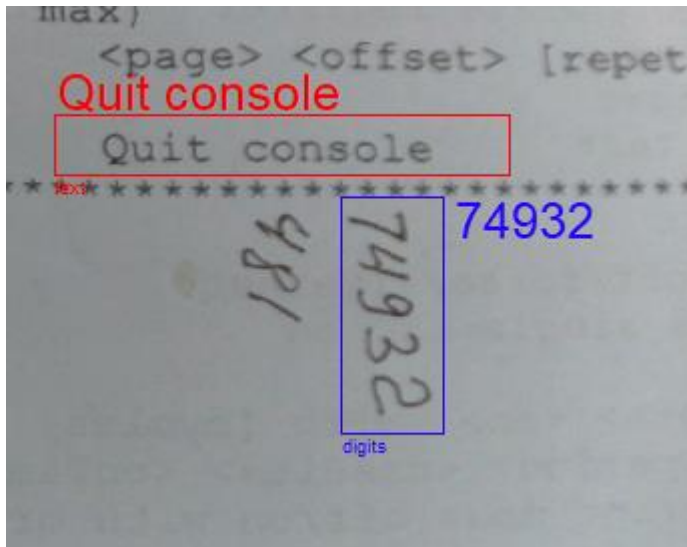
Parameter “**dir**” defines text direction. The values for this parameter could be **right\_3**, **down\_6**, **left\_9** and **up\_0**. Digits are clock hour directions.

Following picture is illustration for text recognition in **right\_3** direction.





Following picture shows simultaneous recognition of text **right\_3** (string “Quit console”) and text **down\_6** (number 74932).



Recognition is working for string slope in several degrees.

Image should be aligned in such way, that string will be vertical or horizontal.

Checkbox **meas\_fast** controls measurement method. It should be unchecked for short (several symbols) strings.

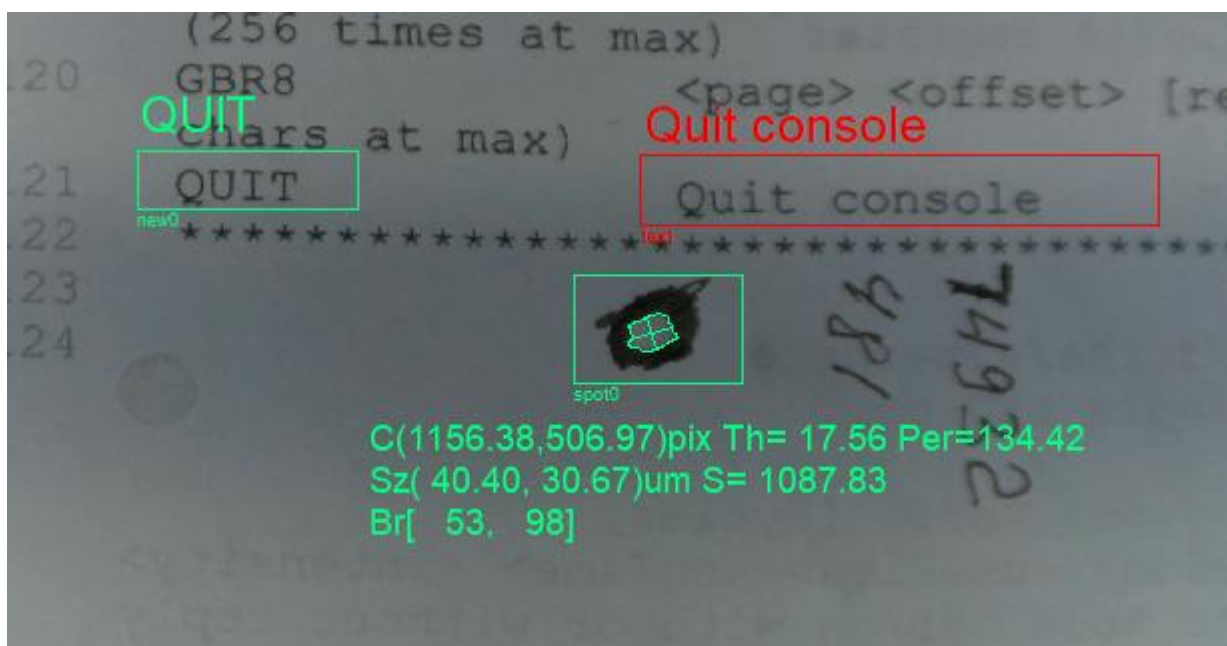
Checkbox **view\_text** controls recognition result view (text could recognized and not shown).

## 1.7. Data flow and formats

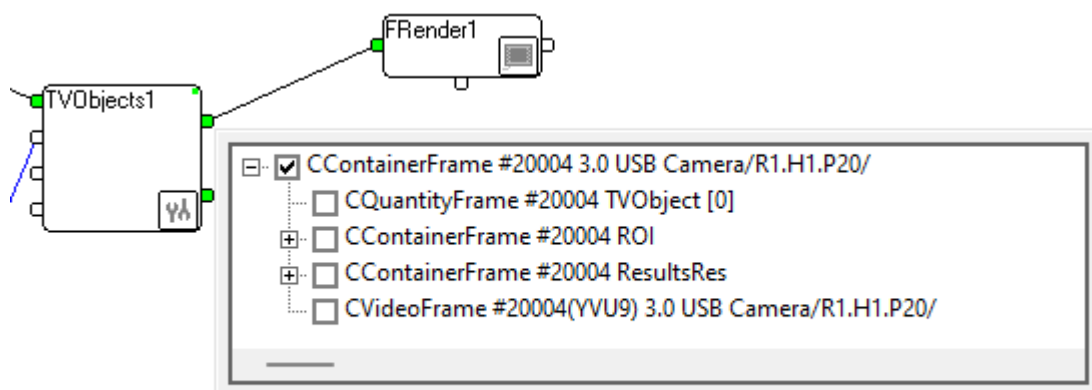
The TVObjects gadget produces container data packet with label "ResultsAll". This container will be consisting of the next packets:

1. Quantity frame with label " TVObjMeas " and value 0. It's necessary for some other gadgets functioning (ScriptRunner gadget, for example)
2. Container with label "ROI" which will hold ROIs of all measured objects. Every ROI is written into figure frame as rectangle.
3. Container with label "ResultRes" with measurement results.
4. Original video frame.

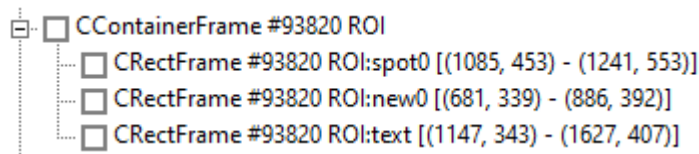
For example, for following image:



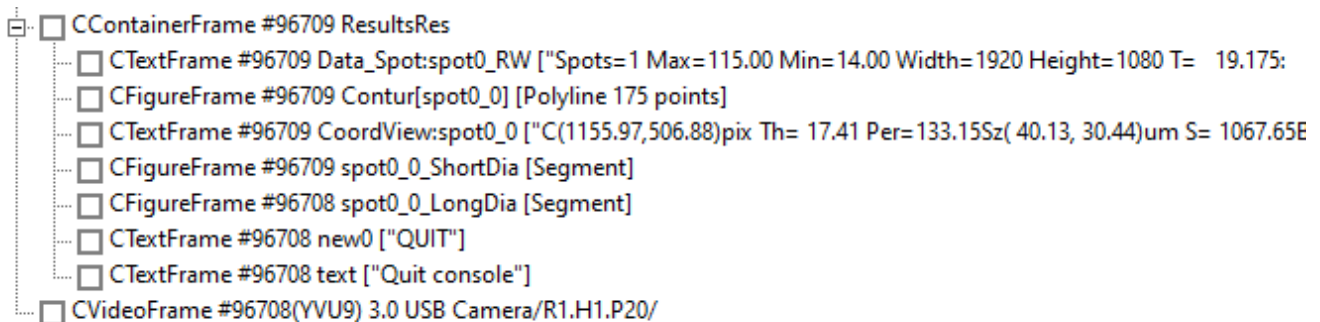
The following data appeared on gadget output.



Detailed ROI container is shown below.



Detailed ResultRes container is below.



The last two frames are results of text recognition: object new0 recognized as “**QUIT**” and object **text** recognized as “**Quit console**”.

The first frame is a result of **spot0** spot measurement.

The second frame is contour around spot (array of points, figure).

The third frame is text for presentation of measurement results on view.

The fourth frame is line for spot long diameter presentation.

The fifth frame is line for spot short diameter presentation.

The first frame (spot measurement result consists of 2 strings:

Statistical result for **spot0**: Spots=1 Max=101.00 Min=6.00 Width=1920 Height=1080 T= 18.863:

Statistics consists of number of found spots, minimal and maximal brightness in analyzed area, video frame size and time from frame appearing in graph to object processing finishing.

Measurement result:

0 1156.38 506.97 46.01 35.84 1088 98 0.0 0.0 17.56 40.40 30.67 0.00 0.00 0.00 0.00 0.00 50 36 95 71 1920 1080 Name=spot0;

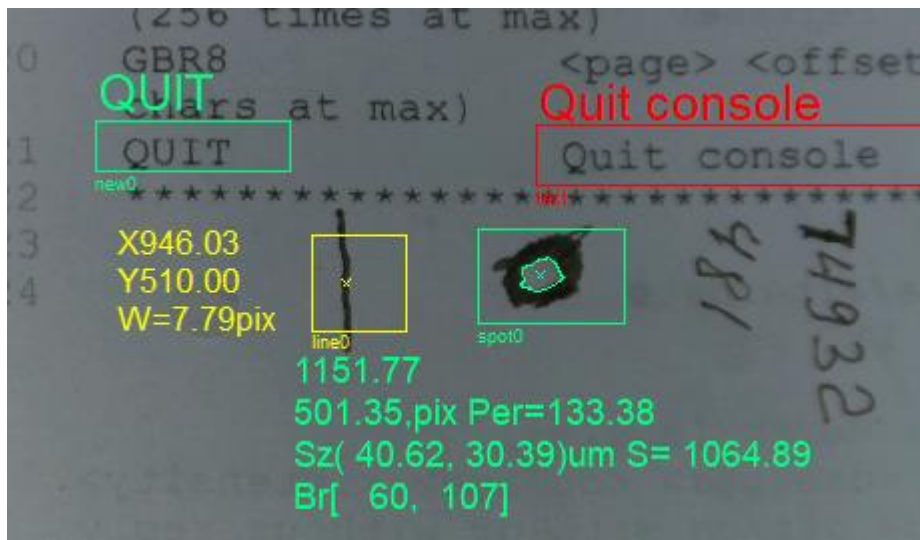
The first number is an index of measured object. If several objects are measured many such strings will be in output frame with \n separator and growing index. Only one spot is measured in this case.

Po s	Description	Value	Note
1	Found object number	0	
2	X Object weight center	1156.38	Relatively to left-top corner
3	Y Object weight center	506.97	Relatively to left-top corner
4	Spot width, pix	46.01	
5	Spot height, pix	35.84	
6	Spot Area, pix	1088	
7	Max brightness inside spot	98	
8	Sum of central 5x5 pixels	0.0	Not measured
9	Number of pixels over thres.	0.0	Not measured
10	Spot angle, degree	17.56	Counter Clock wise direction
11	Long diameter, pix	40.40	
12	Short diameter, pix	30.67	
13	Energy Right	0.00	Diffraction (not measured)
14	Energy Left	0.00	Diffraction (not measured)
15	Energy Top	0.00	Diffraction (not measured)
16	Energy Bottom	0.00	Diffraction (not measured)
17	Energy Center	0.00	Diffraction (not measured)
18	Left edge of out bond rect.	50	Inside search area
19	Top edge of out bond rect.	36	Inside search area
20	Right edge of out bond rect.	95	Inside search area
21	Bottom edge of out bond rect.	71	Inside search area
22	Object name	spot0	

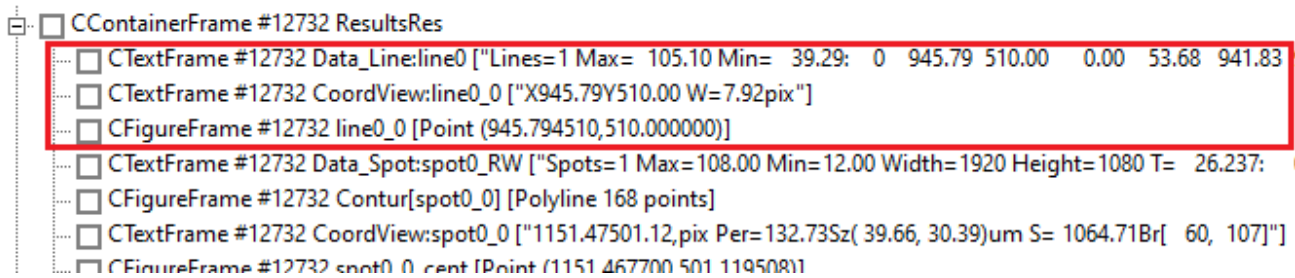
If **Caption Period** field is not zero, periodically short description of parameters (Caption) will be inserted between statistics and data about measured spots.

//Spot#, Xc, Yc, Width, Height, Area, MaxInten, CentSum, OverThr, Angle, Dlong, Dshort, EnergyR, EnL, EnT, EnB, EnC, Left, Top, Right, Bottom, FrWidth, FrHeight;

Following picture includes result of vertical line measurement (marked by yellow).



Output for this measurement shown on next image (marked by red rectangle).



Result of line measurement marked by red. There are 3 frames:

- Main data about measurement results as text
- Text frame for on screen show
- One point figure frame for line center position marking.

Main data holds 2 strings separated by \n.

The first consists of measurement statistics (number of found lines, min and max profile brightness and measurement finish time relatively to input video frame time):

Lines=1 Max = 105.46 Min= 39.10:

The second is one line measurement result:

0 945.50 510.00 0.00 43.84 941.60 949.39 0.00 0.00 16 108 Name=line0;

One such string will be in output for each found and measured line

Format of this string as following:

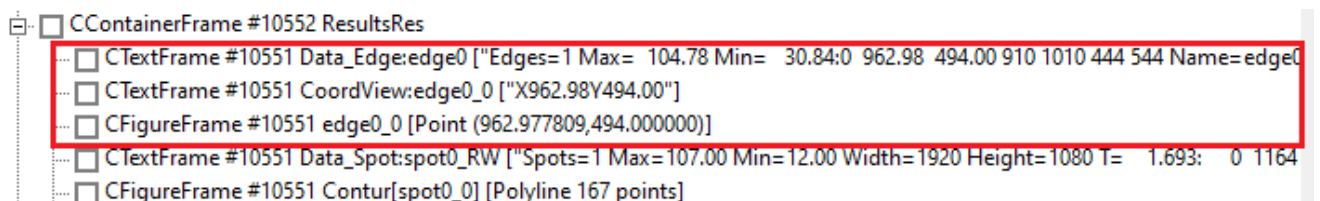
Po s	Description	Value	Note
1	Found object number	0	
2	X Object weight center	945.50	Relatively to left-top corner

3	Y Object weight center	510.00	Relatively to left-top corner
4	Extremal amplitude	0.00	
5	Average amplitude in center	43.84	
6	Left edge of out bond rect.	941.60	
7	Right edge of out bond rect.	949.39	
8	Top edge of out bond rect.	0.00	For vertical line is not measured
9	Bottom edge of out bond rect.	0.00	For vertical line is not measured
10	Minimal brightness in area	16	
11	Maximal brightness in area	108	
12	Object name	line0	

The following image shows edge recognition result.



Output for this measurement shown on next image (marked by red rectangle).



Result of line measurement marked by red. There are 3 frames:

- Main data about measurement results as text
- Text frame for on screen show
- One point figure frame for line center position marking.

Main data holds 2 strings separated by \n.

The first consists of measurement statistics (number of found edges, min and max profile brightness):

Edges=1 Max= 105.21 Min= 34.97:

The second is one edge measurement result:

0 963.98 494.00 910 1010 444 544 Name=edge0;

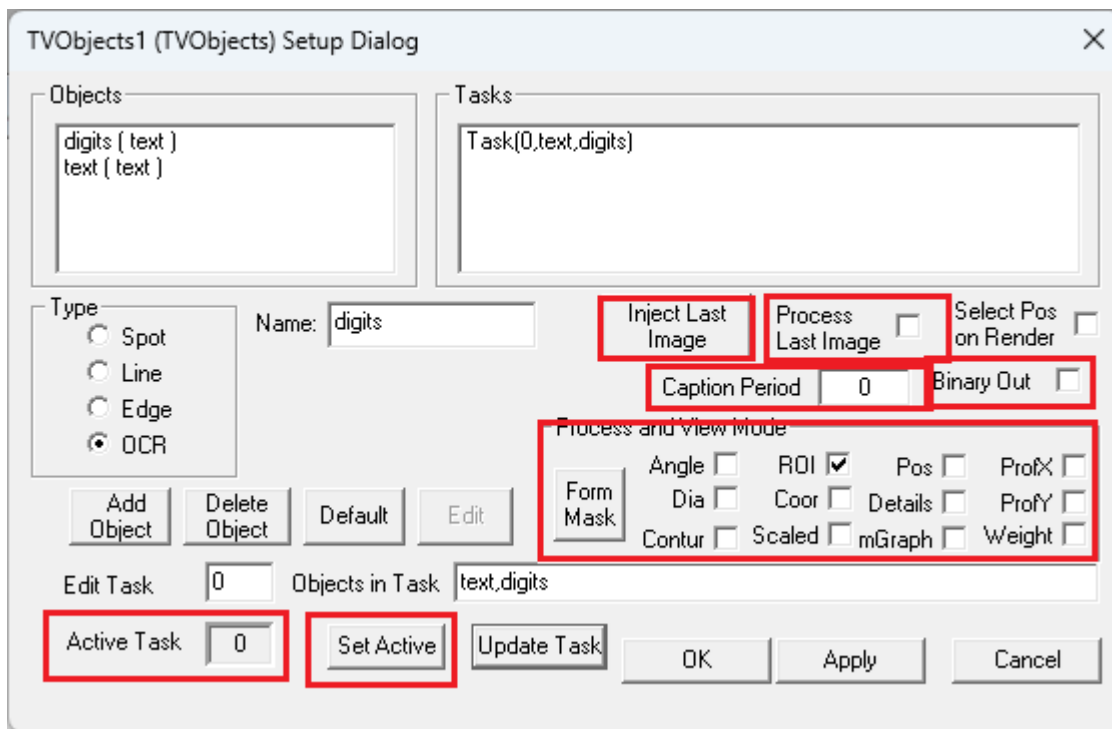
One such string will be in output for each found and measured edge

Format of this string as following:

Pos	Description	Value	Note
1	Found object number	0	
2	X Object weight center	963.98	Relatively to left-top corner
3	Y Object weight center	494.00	Relatively to left-top corner
4	Left edge of out bond rect.	910	
5	Right edge of out bond rect.	1010	
6	Top edge of out bond rect.	444	
7	Bottom edge of out bond rect.	544	
8	Object name	edge0	

## 1.8. Gadget properties.

Gadget has common for all project properties.



- 1.8.1. **Active task** number is task (set of objects measurement) which is used for every image arrived on input. This task could be controlled manually on dialog by task selection from task list (by double click) and click on button **Set Active**.

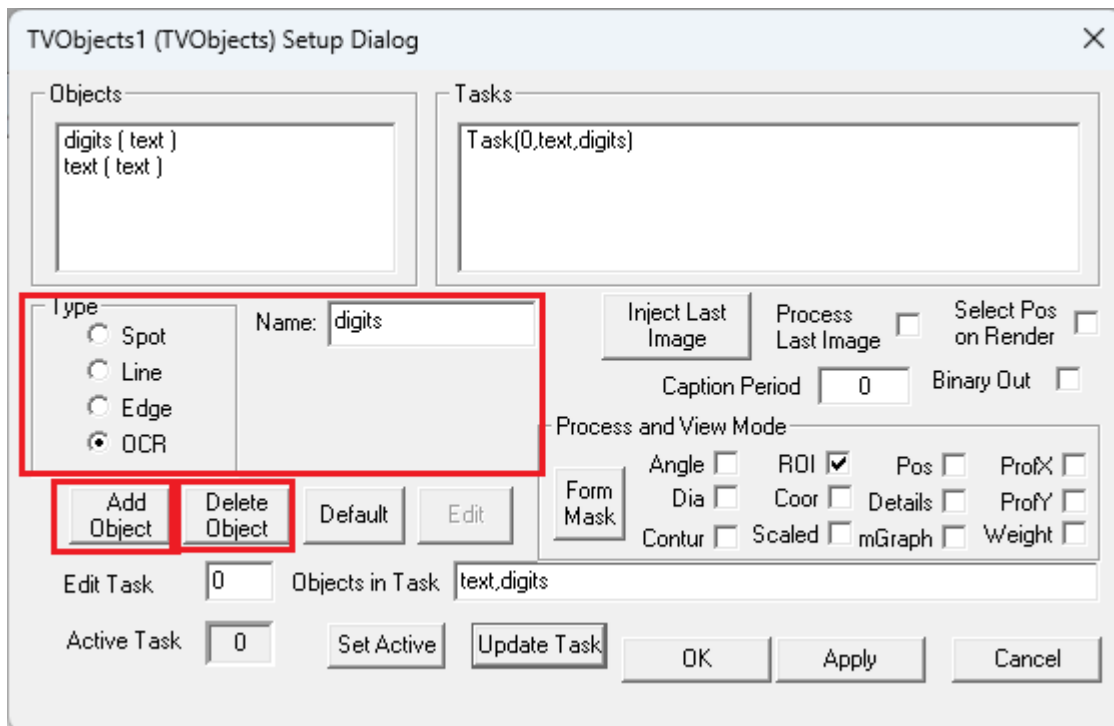
**Active task** could be controlled by text frame arrived to pin 2. Content of this frame should be text “Task(<task number>);”. If <task number> exists in task list, the <task number> become active.

- 1.8.2. Process and view mode checkboxes. These checkboxes are working like view checkboxes for video objects, but applied to all results. For example, checkbox “ROI” enables or disables showing of all ROIs (region of interest) for all objects.
- 1.8.3. **Inject Last Image** button allows to repeat last image. If checkbox **Process Last Image** is checked, this image will be processed with current parameters. For example, if parameters of some video object changed and **Inject Last Image** is pressed, this object will be processed with new parameters.
- 1.8.4. Gadget could send descriptions (caption for result strings) to output. This description could be saved together with results. **Caption Period** value defines how many results strings are between two caption strings. If this value is zero, no caption strings will be sent.
- 1.8.5. **Binary out** checkbox controls output format. Usually, output data provided in text format. The same data could be transferred in binary format for processing time saving.

## 1.9. New video object creation.

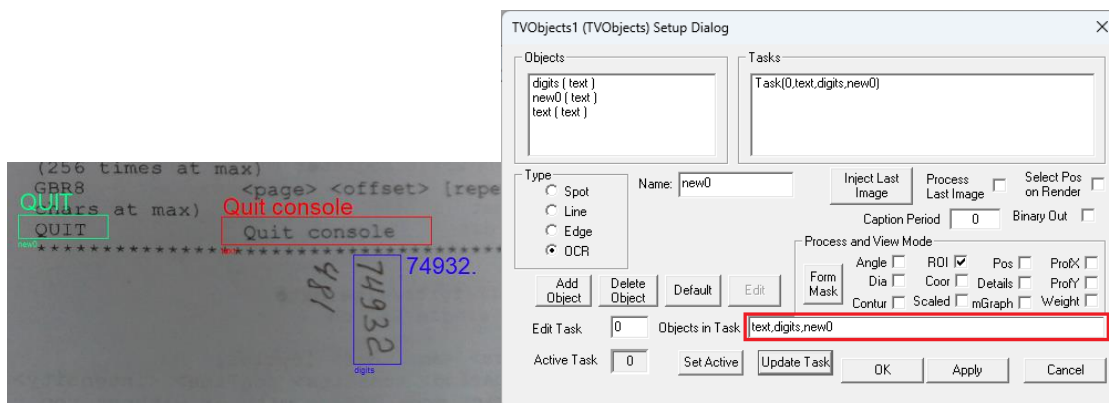


New video object could be created by several control elements on TVObjects setup dialog.



The name of new object should be typed in **Name** field. Object type should be selected. And button **Add Object** should be pressed. After that the new object name appears in **Objects** list and dialog with new object parameters appears. New object name could be added to **Objects in Task** list and sequentially press buttons **Update Task** and **Set Active**. After that TVObjects gadget will process new object for each arrived image with **default parameters set**.

After that the user should change new object parameters for proper processing. The best way for debugging is live video stream generation on gadget input. Following picture shows new text recognition to previous task.



Object could be deleted from list of objects by object selection in this list and click on “Delete Object” button.

## 1.10. Control of objects parameters by text frames.

When gadgets receive on control pin (pin 2 or marked "<<1") a text frame with label "SetObjectProp", this frame will be used for changes in video object parameters. Content of this frame will be treated as property kit (sequence of <Property Name>=<Property Value> separated by semicolon ';'). Processing will be done in following order:

- 1.1.1. Property "name" will be extracted and value will be used for object selection for changes I properties.
- 1.1.2. If object found the text frame content will be sent to object for parameters values replacement (to "ScanProperties" function).
- 1.1.3. If object is not found, nothing will be done.

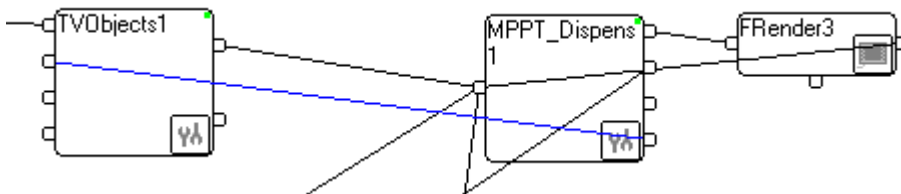
As example, following C++ function could be used for object placement alternation:

```
// Adjust object placement in imaging
bool MPPT_Dispens::SetObjectPlacement( LPCTSTR pVideoObjectName ,
    CRect& ROIPosition )
{
    FXString ROIData;
    // Form command string
    ROIData.Format( "name=%s;xoffset=%d;yoffset=%d;width=%d;height=%d;" ,
        pVideoObjectName ,
        ROIPosition.Left , ROIPosition.top ,
        ROIPosition.Width(), ROIPosition.Height() ); // This is for classic rectangle
        // Usually, width is 'right', height is 'bottom', and lasts parameters
        // should be changed correspondingly

    // send command string to TVObject gadget
    return SetParametersToTVObject( ROIData );
}

bool MPPT_Dispens::SetParametersToTVObject( LPCTSTR pParams )
{
    CTextFrame * pCommand = CreateTextFrame( pParams ,
        "SetObjectProp" );
    return PutFrame( GetOutputConnector( MPPDO_Measurement_Control ) , pCommand );
}
```

Graph fragment for these functions using shown below.



Blue line passes control frames from MPPT\_Dispens1 gadget to TVObjects1 gadget.

## 1.11. Additional possibilities for TVObjects gadget control by text frames.

Some gadget common parameters are controllable by text frames arrived on control pin (pin 2 or marked "<<1"). This text frames **should not have** label "SetObjectProp", which is used for video objects parameters changing (look previous p.11.1).

#### 1.11.1. Scale control

If arrived string holds pattern "Scale(<number>)" or "Scale(<number>,<string>)", the common scale for length or size calculation will be changed. Number will be used as scale value. If string exists, this string will be used for length or sizes units name, otherwise units will be shown as "pix".

For example, if text frame content is "**Scale(1.5632,um)**" the gadget scale will be switched to 1.5632 and all scaled values will be followed by word "um".

#### 1.11.2. Task switch

If arrived string holds pattern "Task(<task number>)" correspondent task will be used for image processing. If task is not exists, no processing will be performed.

#### 1.11.3. View mode switch

If arrived string holds pattern "ViewMode(<View mode mask>)" assigned view mode will be used. This functionality provides control for how result image will be shown on renderer.

View mode mask is binary mask. It can be represented as decimal, hex (in form 0x...) or binary (in form 0b...) number.

Bits are as following

Defines for graphics viewing	Bit Mask	View
#define OBJ_VIEW_POS 0x00000001	// 0 0x0001	Cross on measured position
#define OBJ_VIEW_DET 0x00000002	// 1 0x02	Measurement details
#define OBJ_VIEW_ANGLE 0x00000010	// 4 0x0010	Angle show
#define OBJ_VIEW_SCALED 0x00000100	// 8 0x0100	Show coordinates as scaled
#define OBJ_VIEW_DIFFR 0x00000200	// 9 0x0200	Diffraction areas
#define OBJ_VIEW_CONT 0x00000400	// 10 0x0400	Contour show
#define OBJ_VIEW_PROFX 0x00001000	// 12 0x1000	Profile X
#define OBJ_VIEW_PROFY 0x00002000	// 13 0x2000	Profile Y
#define OBJ_VIEW_DIA 0x00004000	// 14 0x4000	dia show
#define OBJ_WEIGHTED 0x00020000		
#define OBJ_VIEW_TEXT 0x00040000		
#define OBJ_VIEW_CSV 0x00080000	// form data text with comma	
#define OBJ_OUT_MOMENTS 0x00100000	// Do output of image moments in separate text frame	
#define OBJ_VIEW_MRECTS 0x20000000		
#define OBJ_FIND_OBJ 0x10000000		
#define OBJ_VIEW_COORD 0x40000000	// 30 0x40000000	Measured coordinates
#define OBJ_VIEW_ROI 0x80000000	// 31 0x80000000	ROI (search area)