RSB GStreamer Integration - Bug #1057 record ogv can only record Image scopes??

07/04/2012 07:01 PM - J. Sanchez Riera

Status:	Resolved	Start date:	07/04/2012
Priority:	Normal	Due date:	
Assignee:		% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:			

07/04/0040

Description

I'm trying to make a video with images coming from a different scope ex. /nao/vision/LeftRect and I have this error:

nao@pollux:~/software/rsb-gstreamer/branches/0.6/scripts\$./record_ogv.sh ~/Data/ogv/av_fusion/test1_result.ogv /nao/vision/LeftRect

Using video informer scope "/nao/vision/LeftRect"

Using audio informer scope ""

Starting GStreamer Pipeline; Stop with Ctrl-C

Setting pipeline to PAUSED ...

Pipeline is live and does not need PREROLL ...

Setting pipeline to PLAYING ...

New clock: GstSystemClock

Caught SIGSEGV accessing address (nil)

Spinning. Please run 'gdb gst-launch 17649' to continue debugging, Ctrl-C to quit, or Ctrl-\ to dump core.

^CCaught interrupt -- handling interrupt.

Interrupt: Stopping pipeline ...

EOS on shutdown enabled -- Forcing EOS on the pipeline

Waiting for EOS...

The type is an IplImage.

Did you try to record with scopes different than /nao/vision/{0,1}?

History

#1 - 07/09/2012 04:46 PM - J. Wienke

- Description updated

This looks like you are sending unexpected data on the scope. Can you please let a logger run on the respective scope and verify that there are no unexpected events.

#2 - 07/10/2012 05:15 PM - J. Sanchez Riera

No. I think the data sent by the scope was correct. I rerun the experiments and this is the output of the logger

bytearray is the only type send in this scope

nao@pollux:~/software/rsbvideoreceiver/trunk/build\$ rsb-loggercpp0.6 --scope /nao/vision/LeftRect --style detailed

Event

04/24/2024 1/4

Scope /nao/vision/LeftRect/

Id EventId[participantId = UUID[9402b2ab-fc10-4718-b2a9-9205cd7158a2], sequenceNumber = 3468]

Type bytearray

Origin 9402b2ab-fc10-4718-b2a9-9205cd7158a2

Timestamps

Create 2012-Jun-14 14:39:24.123673+??:??

Send 2012-Jul-10 17:10:54.364174+??:??

Receive 2012-Jul-10 17:10:54.367184+??:??

Deliver 2012-Jul-10 17:10:54.367187+??:??

Causes

EventId[participantId = UUID[f436cbbd-fa64-4297-8df2-a39dd9b6e941], sequenceNumber = 577]

Payload (bytearray, length 307216)

#3 - 07/10/2012 05:23 PM - J. Wienke

Oh, sorry, you have to do this with the common lisp logger with rst injected. The C++ logger says bytearray to everything it does not know. The cl version can be donwloaded from the CI server.

#4 - 07/10/2012 05:24 PM - J. Wienke

So far I can however say that an image containing only zeros is very unlikely. So I suspect something else is sent there.

#5 - 07/10/2012 05:24 PM - J. Wienke

Another test could be if you can receive the images sent there with the rsbvideoreceiver.

#6 - 07/10/2012 05:31 PM - J. Sanchez Riera

I'll look at this cl logger.

I did the check with rsbvideoreceiver and it displays the image correctly. The zeros are not suspicious, because it's showing only the first bytes of the rectified image. (In our rectified image corners are black because image is reshaped).

#7 - 07/10/2012 05:33 PM - J. Wienke

Ok, so maybe what you can also do is rerun gstreamer again and generate the core dump using the keyboard shortcut given in the gstreamer output. Please ensure before that the gstreamer plugins are compiled using -DCMAKE_BUILD_TYPE=debug so that debug output will be meaningful.

#8 - 07/11/2012 06:10 PM - J. Sanchez Riera

This is the output with the cl logger:

04/24/2024 2/4

Event

Scope : /nao/vision/LeftRect/

ld : 6F486BA2-F3A3-535D-AB48-A3178F79B44F

Sequence-Number: 1156

Origin : 4B09AFE0-719E-4188-8B78-315D26B29352

Method : N/A

Timestamps

Create: 2012-06-14T14:39:24.123673+02:00 Send: 2012-07-11T18:04:55.088108+02:00 Receive: 2012-07-11T18:04:55.090575+02:00 Deliver: 2012-07-11T18:04:55.090581+02:00

Meta-Data

Rsb.Transport.Payload-Size: 307216

Rsb.Transport.Wire-Schema:.rst.vision.Image

Causes

644E0B1F-78A5-5885-9C45-6D4A22F8B330

Payload (SB-KERNEL::SIMPLE-ARRAY-UNSIGNED-BYTE-8)

... [omitting 307,120 of 307,216 octets]

I cannot see any other type than .rst.vision.Image

About the ./record_ogv I could not manage to get debug info. I compiled rsb-gstreamer with the flag as you indicated, I get a much heavier libgstrsb.so, I copy it to ./gstreamer-0.10/plugins, but when I execute the script I get exactly the same useless output.

#9 - 07/11/2012 07:44 PM - J. Wienke

Ok, so pressing ctrl+/ does not produce anything more?

Can you please record a SHORT tide file that demonstrates the problem and put it on the webday, so that we can debug it here on our own?

#10 - 07/16/2012 11:30 AM - J. Sanchez Riera

https://webdav.humavips.eu/pub/test.tide

here there's a sample file.

#11 - 07/18/2012 03:54 PM - J. Moringen

We tried to reproduce the problem using the linked tide file, but could not find any events on the /nao/vision/LeftRect/ scope. Should these be include in the tide file or are they generated by a separate component? In the latter case, could you provide another tide file including these events?

Apart from that, the record_ogv.sh script has to be used slightly differently when working without audio:

\$./record_ogv.sh ~/Data/ogv/av_fusion/test1_result.ogv_/nao/vision/LeftRect

04/24/2024 3/4

Note the _ at the end. This could already solve the problem, but we cannot currently verify this because of the incomplete tide file.

#12 - 07/18/2012 04:25 PM - J. Sanchez Riera

Yes it's true, /nao/vision/LeftRect is not included in that tide file. You can use test_vasil.tide instead. Is in the same place (webdav pub). Unfortunatelly this slightly different way to call the script didn't change anything.

#13 - 07/18/2012 06:56 PM - J. Moringen

We used test_vasil.tide for further tests and found that our GStreamer element cannot handle the image format which is

Width: 640
Height: 480
Channels: 1

Depth: DEPTH-8U
Color-Mode: COLOR-BGR

Data-Order: DATA-INTERLEAVED

We have updated the GStreamer element so that it can handle the images if the format is changed to

Width: 640
Height: 480
Channels: 1

Depth: DEPTH-8U

Color-Mode: COLOR-GRAYSCALE Data-Order: DATA-INTERLEAVED

This may be a bug in the IpIImageConverter class.

#14 - 07/18/2012 07:06 PM - J. Wienke

To clarify this: your component which produces the rectified images, declares an illegal format in the RST image type, because BGR images need 3 channels and not one. Which component is this and how does it generate rst.vision.Image data? On your side you need to fix this output format issue to produce valid images.

#15 - 07/22/2012 07:03 PM - J. Moringen

- Status changed from New to Resolved
- % Done changed from 0 to 100

See r66.

04/24/2024 4/4