

AnalyticVideo Camera & AnalyticVideo Tracker
For Axis Network Cameras
AXIS Viewer Guide

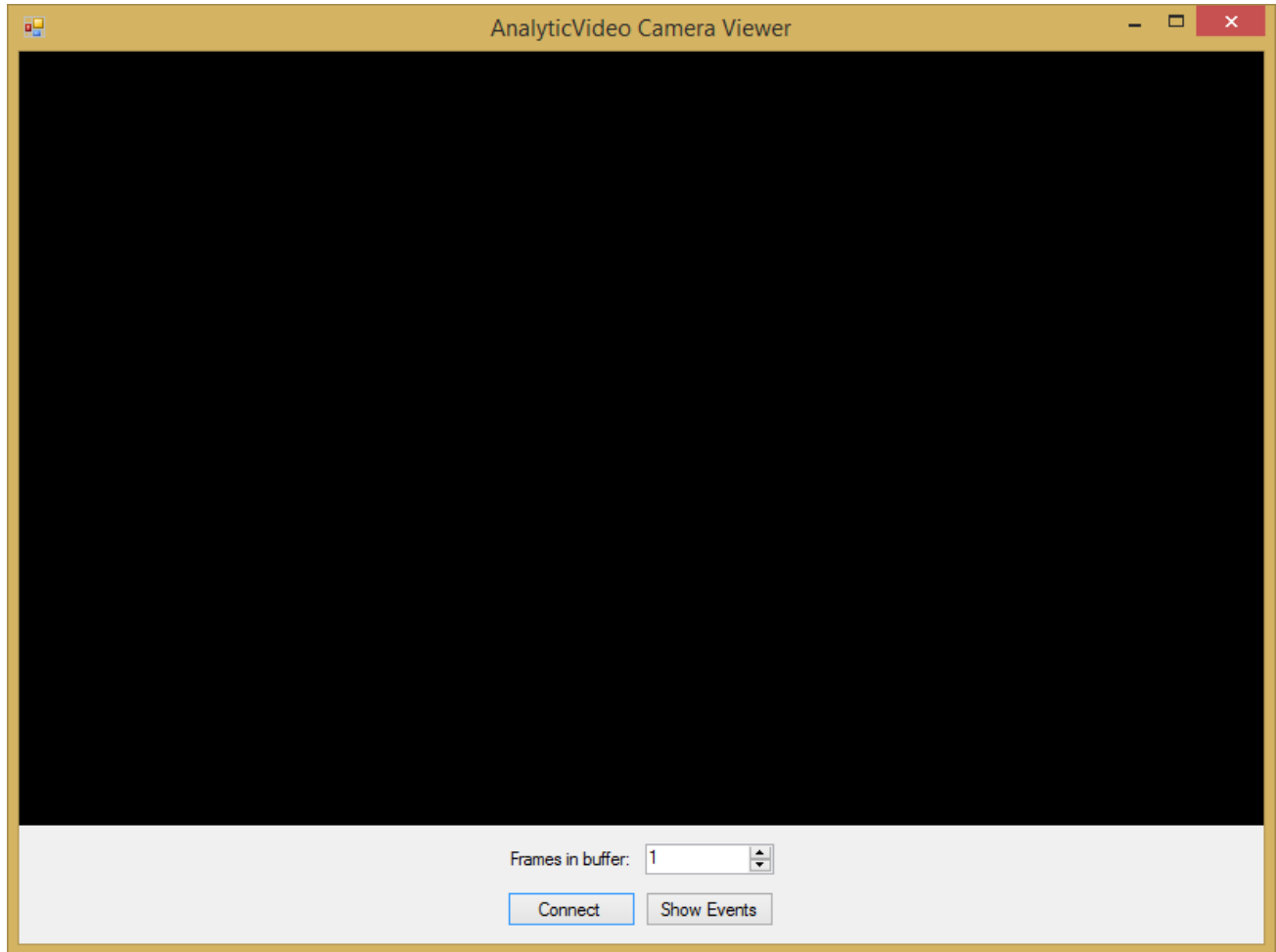
Copyright © 2014 Institute of Digital and Optical Technologies B. V.

All rights reserved

Contents

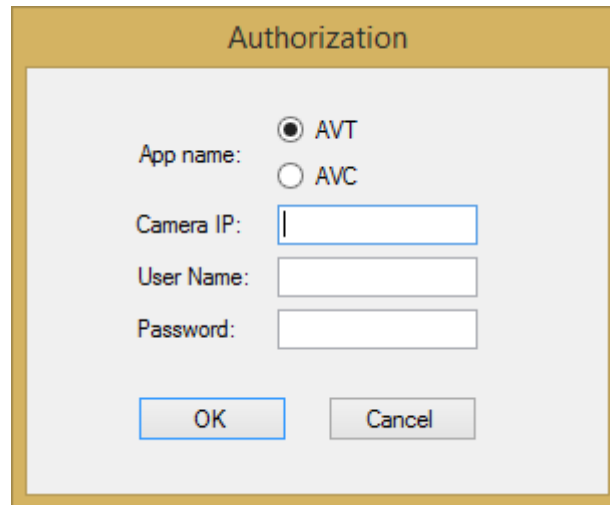
| | |
|--|-----------|
| CONTENTS..... | 2 |
| USER INTERFACE | 3 |
| AXIS SDK METHODS USED IN AVCVIEWER..... | 6 |
| CONTACT US | 10 |

USER INTERFACE



Pic. 1. Main application window.

The program interface includes an area to display images from the camera and a group of controls. To connect to the camera press the “Connect” button.



Pic. 2. Authorization dialog.

Authorization dialog with camera IP address will appear. First, you must choose the name of the plugin. Then you have to fill in camera user name and password. If AVT or AVC plugin is running on the camera and the entered data is correct, the connection will be established and a real-time video will be displayed. In case of an error a message: "Can't connect to camera" will pop up.

Depending on the chosen resolution in the plugin settings, as well as on the camera type, objects data, received from the application, can be asynchronous with the received image. Therefore, in case of visual lag of the frames around objects, the number of buffered frames can be increased from 1 up to 60. To do this change the value of the "Frames in Buffer" control.

| | Time | X | Y | Width | Height | Object ID | Object Type | Human Count |
|----|------|---|---|-------|--------|-----------|-------------|-------------|
| ▶* | | | | | | | | |

Pic. 3. Events viewer.

To display the events received from the camera, press the “Show Events” button, which opens a dialog box containing a list of all received events since the moment of connection.

AXIS SDK METHODS USED IN AVCVIEWER

AXIS Media Parser SDK 3.62 is used to display images from the camera in the real time.

Used interfaces:

AxisMediaParser

Methods:

```
public: void Connect(
    [out] LONG* pCookieID,
    [out] LONG* pNumberOfStreams,
    [out] VARIANT* pMediaType
);
```

It establishes a connection to an Axis device and returns media information about the specified stream to the caller.

```
public: void Start(
    void
);
```

It instructs AXIS Media Parser to start the network stream. Once the stream has been started, AXIS Media Parser will start extracting media samples and trigger data from the stream.

```
public: void Stop(
    void
);
```

It instructs AXIS Media Parser to stop the network stream. Once the stream has been stopped, AXIS Media Parser will stop extracting media samples and trigger data from the stream.

Events:

```
public: void OnVideoSample(
    [in] LONG CookieID,
    [in] LONG SampleType,
```

```

    [in] LONG SampleFlags,
    [in] ULONGLONG StartTime,
    [in] ULONGLONG StopTime,
    [in] VARIANT SampleArray
);

```

It will be called for each video frame received.

```

public: void OnError(
    [in] LONG ErrorCode
);

```

It will be called for each error caused by connection problems.

AxisMediaViewer

Methods:

```

public: void Init(
    [in] LONG NumberOfStreams,
    [in] VARIANT MediaType,
    [in] LONGLONG hWnd
);

```

It initiates the AXIS Media Viewer with data from the stream.

```

public: void Start(
    void
);

```

It instructs the AXIS Media Viewer to start decoding media samples.

```

public: void Stop(
    void
);

```

It instructs the AXIS Media Viewer to stop decoding media samples.

```
public: void RenderVideoSample(
    [in] LONG SampleFlags,
    [in] ULONGLONG StartTime,
    [in] ULONGLONG StopTime,
    [in] VARIANT SampleArray
);
```

It renders a video frame. It will wait until previous sample is processed.

Events:

```
public: void OnDecodedImage(
    [in] ULONGLONG StartTime,
    [in] SHORT ColorSpace,
    [in] VARIANT SampleArray
);
```

It will be called for every decoded image frame.

To obtain an analytic data from the camera it is necessary to create a new `WebClient` object, set authorization data and register a callback function `WebClient::UploadValuesCompleted`, which is called when a response from the camera is received. Then an asynchronous HTTP POST request `WebClient::UploadValuesAsync` is sent to the URL “`http://<cameraIP>/local/<plugin_name>/analyze.cgi`”.

```
wb = new WebClient();
wb.UseDefaultCredentials = true;
wb.Credentials = new NetworkCredential(authForm.UserName,
authForm.Password);
wb.UploadValuesCompleted += CompleteResponse;
var data = new NameValueCollection();
```


To obtain image frames from the camera it is necessary to create two AXIS Media Parser/Viewer COM objects. After that, user name, password and the camera stream URL (“http://< cameraIP>/axis-cgi/mjpg/video.cgi?resolution=<width>x<height>”) should be specified by setting fields `AxisMediaParser::MediaUsername`, `AxisMediaParser::MediaPassword` and `AxisMediaParser::MediaURL`.

To get <width> and <height> values it is necessary to send HTTP POST request `WebClient::UploadValues` to the URL “http://<cameraIP>/local/<plugin_name>/get_resolution.cgi”. If the response message is blank, the camera resolution was not set yet and the default resolution 640x480 is used. Otherwise, the response will contain a line like <width>x<height>. Next, it is necessary to register a callback function `AxisMediaParser::OnVideoSample`, which is called when a new video frame is received.

The next step is to configure an `AxisMediaViewer` object. To do this register a callback function `AxisMediaViewer::OnDecodedImage`. It is called when a new frame is decoded. Fields `AxisMediaViewer::EnableOnDecodedImage` and `AxisMediaViewer::LiveMode` should be set to true.

After that, it is necessary to call the `AxisMediaParser::Connect` function to connect to the camera and the `AxisMediaViewer::Init` function to initialize the decoding process.

The final action is to launch processes of image obtaining and image processing `AxisMediaViewer::Start` and `AxisMediaParser::Start`.

CONTACT US



Support: support@analyticvideo.com

Licensing: bizdev@analyticvideo.com

Web: www.AnalyticVideo.com

i.Dot

Headquarters

Institute of Digital and Optical Technologies B. V.

Postal address: De Lairessestraat 160-C, 1075 HM
Amsterdam, the Netherlands

Fax: +31.20-8908634

E-mail: bizdev@idotbv.nl

Web: www.idotbv.nl

Copyright © 2014 Institute of Digital and Optical Technologies B. V. All rights reserved.

This guide is for informational purposes only. No warranties, express, implied or statutory, as to the information in this document. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Institute of Digital and Optical Technologies B. V.