Zebra
Aurora™ Vision

Aurora Vision Studio 5.7

Working with GigE Vision® Devices

Created: 11/10/2026

Product version: 5.7.5.80527

Table of content:

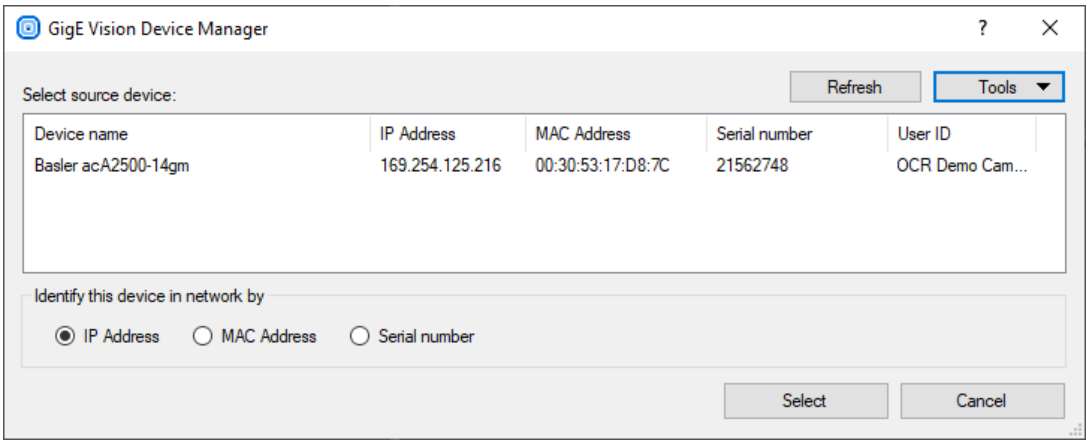
- GigE Vision® Device Manager
- Connecting Devices
- Enabling Traffic in Firewall
- Enabling Jumbo Packets
- Device Settings Editor
- Known Issues

GigE Vision® Device Manager

The GigE Vision Device Manager is available from *Tools* section within the Main Window Menu. It also appears when a user starts to edit some properties of GigE Vision®-related filters.

Device Manager Functions

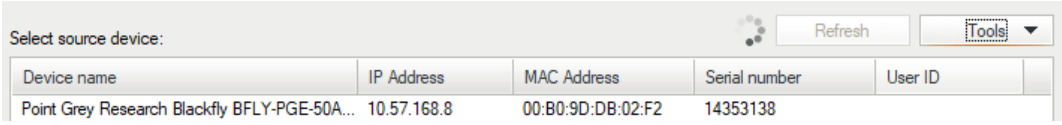
Typical state of the Device Manager is shown on image below. Note that the window may change its appearance depending on its purpose (like selecting a device address in a filter).



At first the manager will search local network for active devices. All found devices will be shown in list with the following information: manufacturer name and device name, current IP address, network interface hardware address (MAC address), serial number (if supported), user specified name (saved in the device memory; if supported by device). Information like MAC address and serial number should be printed on the device casing for easy identification. Sometimes, when a device has more than one interface, it may appear in the list more than once. In this situation every entry in the list identifies another device feature.

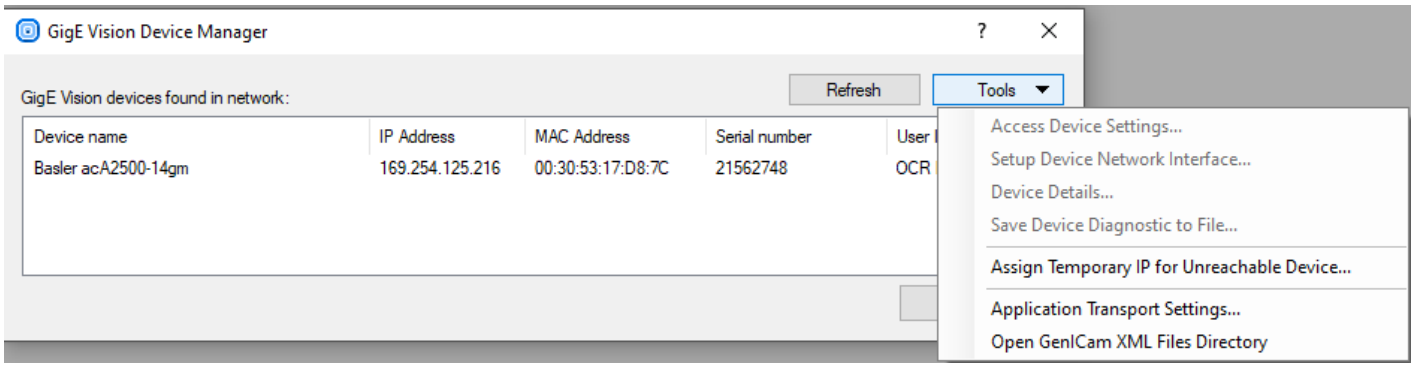
Refresh

Refresh button performs a new search in the network. Use this function when the network configuration has been changed, a new device has been plugged in or when your device has not been found at startup.



Tools

The Tools button opens a menu with functions designed for device configuration. Some of these functions are device-dependent and require the user to select a device on the list first (they are also available in a device context menu).



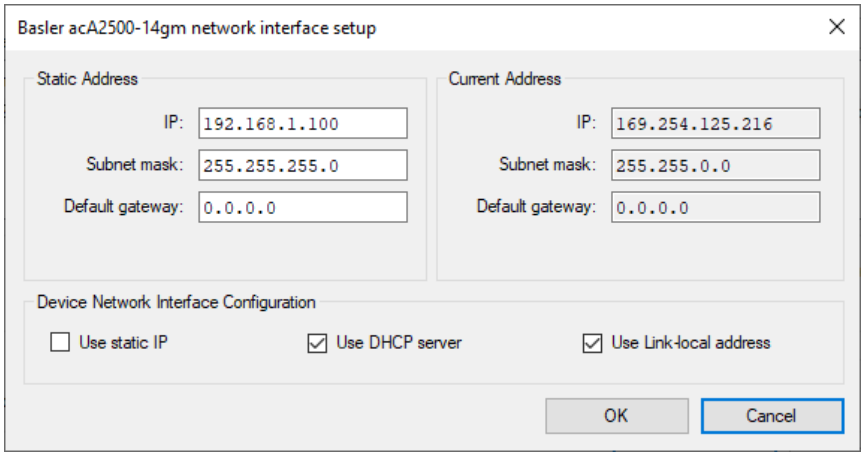
Tool: Access Device Settings...

This tool allows to access device-specific parameters prepared by its manufacturer and available through GenICam interface.

See: Device Settings Editor

Tool: Setup Device Network Interface...

This tool is intended to manage network configuration of a device network adapter.

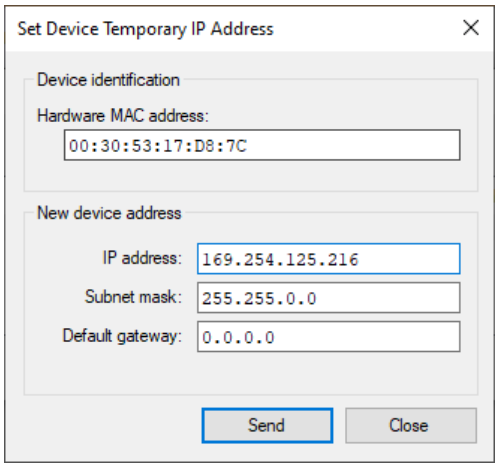


- **Static address** – this field allows to set a static (persistent) network configuration saved in device non-volatile memory. Use this setting when the device is identified by IP address that cannot change or when automatic address configuration is not available. This field has no effect when *Use static IP* field is not checked.
- **Current address** – this read-only field shows current network configuration of a device, for example the address assigned to it by a DHCP server.
- **Device IP configuration** – this field allows to activate or deactivate specified methods of acquiring addresses by a device on startup. Some of this options can be not available (grayed) when the device is not supporting specified mode.
  - **Use static IP** – Device will use address specified in *Static address* field.
  - **Use DHCP server** – When a DHCP server is available in the network, devices will acquire automatically assigned address from it.
  - **Use Link-local address** – When there is no other method available a device will try to find a free address from 169.254.-.- range. When using this method (for example on a direct connection between the device and a computer) the device will take significantly more time to become available in network after startup.

After clicking OK, the new configuration will be sent to a device. Configuration can be changed only when the device is not used by another application and/or is not streaming video. New configuration may be not available until the device is restarted or reconnected.

Tool: Assign Temporary IP for Unreachable Device

This tool is intended for situations when a device cannot be accessed because of its invalid or unspecified network configuration (note that this should be a very rare case and usually the device should appear in list). The tool allows to immediately change network address of an idle device (thus realizing GigE Vision® FORCE IP function).



This tool requires a user to specify device hardware network adapter MAC address (should be printed on device casing). After that a new IP configuration can be specified. The address can be changed only when the device is idle (is not connected to other application and not streaming video). The new address will be available immediately after successful send operation.

Tool: Application Transport Settings...

This tool allows access to and editing of application settings related to the driver transport layer, like connection attempts and timeouts. Settings are saved and used at the whole-application level. Changes affect only newly opened connections.

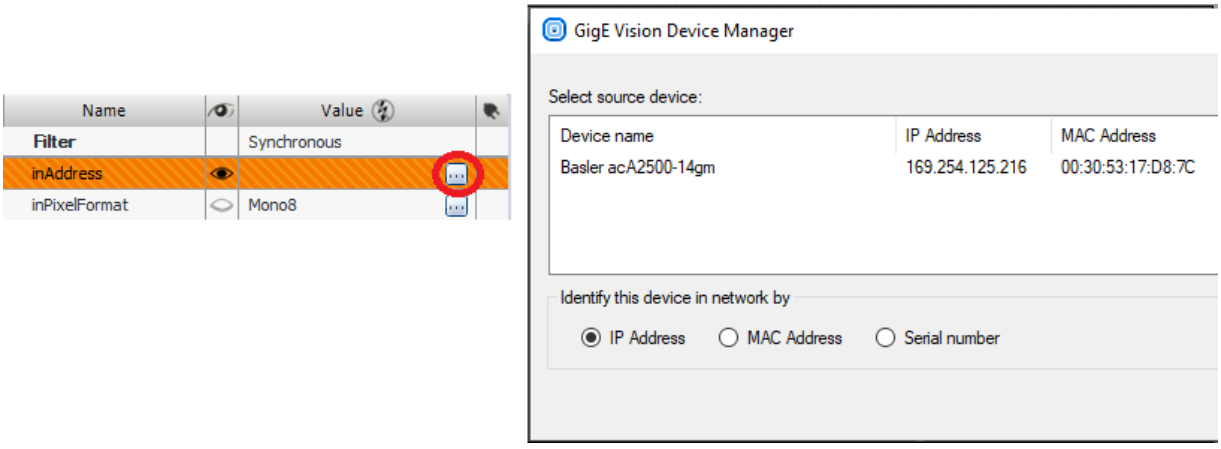
Tool: Open GenICam XML Directory

GigE Vision® devices are implementing GenICam standard. GenICam standard requires that a device must be described by a special XML file that defines all device parameters and capabilities. This file is usually obtained automatically by the application from the device memory or from manufacturer's internet web page. Sometimes the XML file can be supplied by manufacturer on a separate disk. Aurora Vision Studio and Aurora Vision Executor use a special directory for these files which is located in the user data directory. Use this tool to open that directory.

Device description files should be copied into this directory without changing their name, extension and content. File can also be supplied as a ZIP archive – do not decompress such file nor change its extension.

Selecting Device Address for Filter

Device Manager can be opened as a helper tool for editing device address in filter properties.



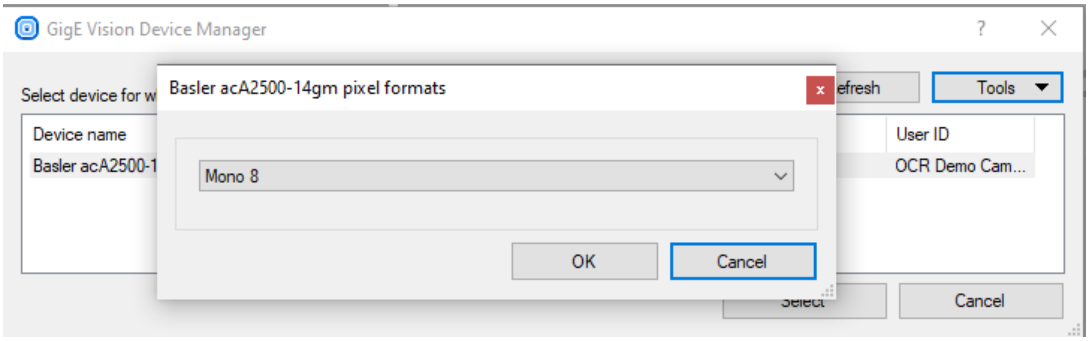
In this mode the manager allows selecting one device from the list (by selecting it on the list and clicking Select or by double-clicking on the list). Below the list, options are shown that determine how a device will be identified in program:

- **IP Address** – (Default) Current device IP address will be saved in program. This mode allows for faster start, but when the device is using automatic IP configuration its address can change in future and device will become unavailable for the program.
- **MAC Address** – Device hardware MAC address will be saved in a program. Upon every program start application will search for the device to obtain most recent IP address. This mode allows to identify device even when its IP is changed.
- **Serial number** – Device serial number will be saved in a program. Use this mode only when device serial number identification is supported by device. Upon every program start application will search for the device to obtain most recent IP address. This mode allows to identify the device even when its IP has changed.

After device selection, address in the chosen format is inserted into a filter's property value.

Selecting Pixel Format

Device manager can be opened as helper tool for editing device pixel format for GigEVision_GrabImage filter parameter.



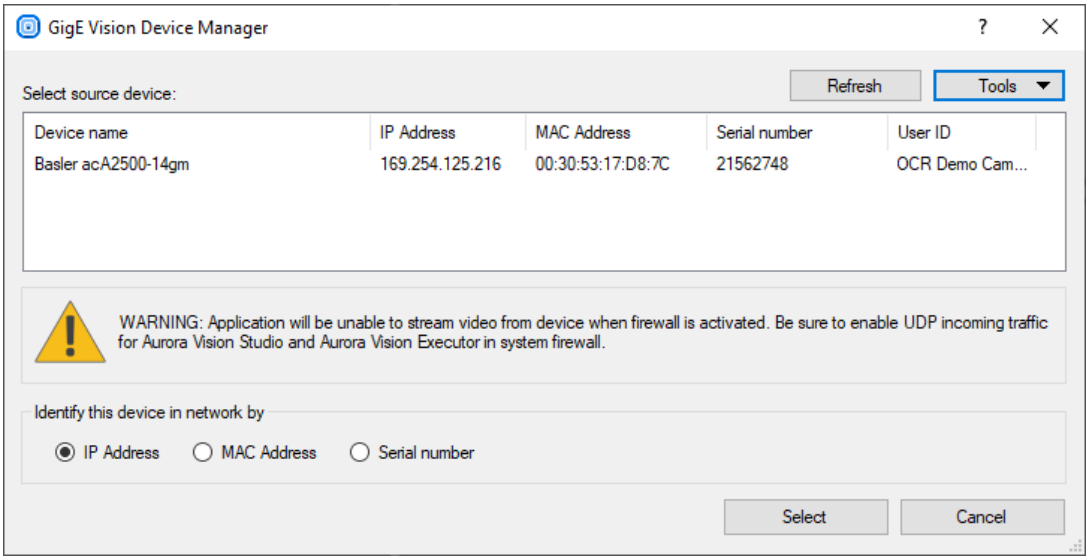
In this mode an additional dialog is shown with a drop-down list of pixel formats supported by a device. When there is more than one device in the network, you must first select the device for which the format will be selected. When there is no device available in network a list of all standard pixel formats will be shown.

Pixel format list contains all formats signaled by a device and uses the name provided by the device. If it is a device-specific format, it may not be possible to decode it.

After selecting a format and clicking Ok the format name will be inserted into a filter's property value.

Firewall Warning

In some situations device manager might show a warning message about Windows firewall.



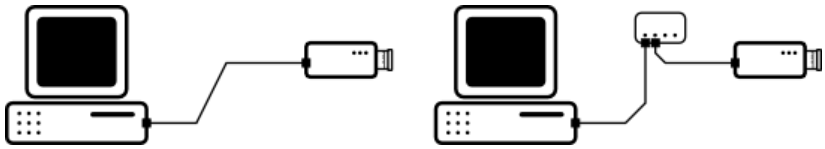
This warning is shown when the application has determined that Windows firewall will block any type of UDP traffic for application. For some devices it is required that incoming UDP traffic is explicitly allowed in firewall. Lack of this warning does not indicate that firewall will not block connection (for example if there is a 3rd party firewall in the system or when the Windows firewall allows traffic in Private domain but a device is in an unidentified network).

To clear out this message you must enable incoming traffic on all UDP ports in Windows firewall. For more information about enabling traffic in Windows Firewall see: Enabling Traffic in Firewall.

Connecting Devices

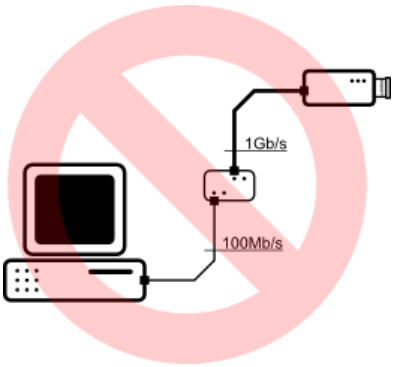
Connecting a GigE Vision device to a computer means plugging both into the same Ethernet network.

It is recommended that the connection is as simple as possible. To achieve best performance use direct connection with a crossed Ethernet cable or connect the camera and the computer to the same Ethernet switch (without any other heavy traffic routed through the same switch).



The device and the computer must reside in a single local area network and must be set up for the same subnet.

GigE Vision® is designed for 1 Gb/s networks, but it is also possible to use 100 Mb/s connection as long as the entire network connection have an uniformed speed (some custom device configuration might be required when the device is not able to detect connection speed automatically). It is recommended however to avoid connecting a device to a network link which is faster than the maximum throughput of the whole network route. Such configurations require manual setting of the device's transmission speed limit.



Firewall Issues

GigE Vision® protocol produces a specific type of traffic that is not *firewall friendly*. Typical firewall software is unable to recognize that video streaming traffic is initialized by a local application and will block this connection. Aurora Vision's GigE driver attempts to overcome this problem using *firewall traversal mechanism*, but not all devices support this.

It is thus required to enable incoming traffic on all UDP ports for Aurora Vision Studio and Aurora Vision Executor in a firewall on your local computer.

For information how to enable such traffic in Windows Firewall see: Enabling Traffic in Firewall.

Configuring IP Address of a Device

In most situations a GigE Vision device is able to automatically obtain an IP address without user action (using DHCP server or automatic local link address). It is however recommended to set a static IP address for both local network card and device whenever possible. In some cases (e.g. when preparing the device for operation in an industrial network) it might be required to access and set/change the device's network configuration for a proper static IP address. Most suitable for this purpose will be a software and a documentation provided by the device manufacturer. When these are not available Aurora Vision Studio offers universal configuration tools available from the GigE Vision Device Manager (see: Device Manager section).

Packet Size

Network video stream is divided into packets of a specified size. The packet size is limited by the Ethernet standard but some network cards support an extension called *jumbo packet* that increases allowed packet size. Because a connection is more efficient when the packet size is bigger, the application will attempt to negotiate biggest possible network packet size for current connection, taking advantage of enabled jumbo packets.

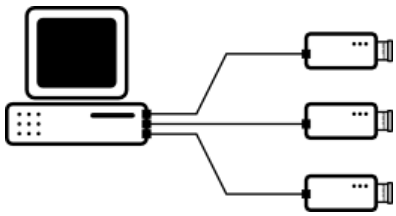
For information how to enable jumbo packets see: Enabling Jumbo Packets

Connecting Multiple Devices to a Single Computer

It is possible to connect multiple GigEVision cameras to a single computer and to perform image processing based on multiple video streams (e.g. observing objects from multiple sides), however it can introduce multiple technical challenges that must be considered.

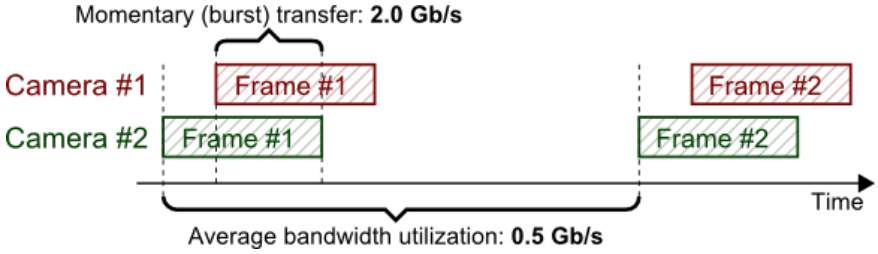
For the best performance it is recommended to connect all devices directly to the computer using multiple

gigabit network cards:



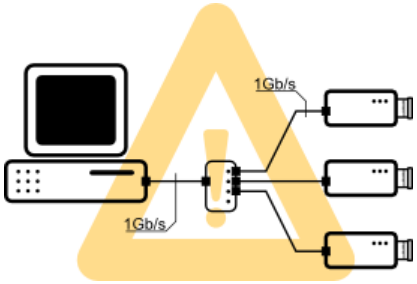
In such configuration it is required for the computer hardware to handle concurrent gigabit streams at once. Even when separate network cards are able to receive the network streams there still may be problems for the computer hardware to transfer data from the network adapters to the system memory. Attention must be paid when choosing hardware for such applications. When given requirements are not met the system may observe excessive packets loss leading to the video stream frames loss.

Even when the cameras framerate is low and the resulting average network throughput is relatively low the system still **may drop packets during network bursts** when the momentary data transfer exceeds the system capabilities. Such burst may appear when multiple cameras transmit a single frame at the same time. By default GigEVision camera is transferring a single frame with maximum available speed and lowering framerate is only increasing the gaps between frame transfers:



Although diagnostic tools will report network throughput utilization to be well below system limits it is still possible for short burst transfers to temporary exceed the system limits resulting in packets drop. To overcome such problems it is required to not only ensure camera framerates to be below proper limit, but also to limit the maximum network transfer speed of the device network adapters. Refer to the device documentation for details about how to limit the network transfer speed in specific device. Usually this can be achieved by decreasing value of parameters such as *DeviceLinkThroughputLimit* or *StreamBytesPerSecond* (in bytes per second), or by introducing delays in between the network packets by increasing parameters such as *PacketDelay*, *InterPacketDelay* or *GevSCPD* (measured in internal device timer ticks – must be calculated individually for device using device timer frequency).

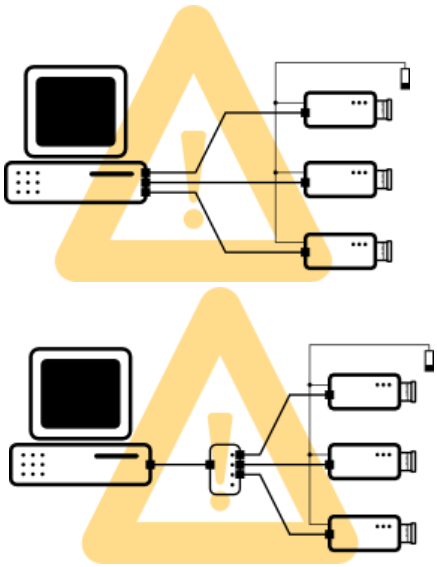
Above requirements are especially important when cameras are connected to the computer using a single network card and a shared network switch:



Special care must be taken to assure that all the cameras connected to the switch (when all transmitting at once) do not exceed transfer limits of the connection between the switch and the computer, both average transfer (by limiting the framerate) as well as temporary burst transfer (by limiting network transfer speed). Network switch will attempt to handle burst transfers by storing the packets in its internal buffer and transmitting packets stored in the buffer after the burst, but when the amount of data in burst transfer exceeds the buffer size the network packets will be dropped. Thus it is required for the switch buffer to be large enough to store all camera frames captured at once, or to limit transmission speed for the switch buffer to not overflow.

It is important to note that **the maximum performance of the multi-camera system with shared network switch is limited by the throughput of the link between the switch and the computer**, and usually it will not be possible to achieve the maximum framerate and/or resolution of the cameras.

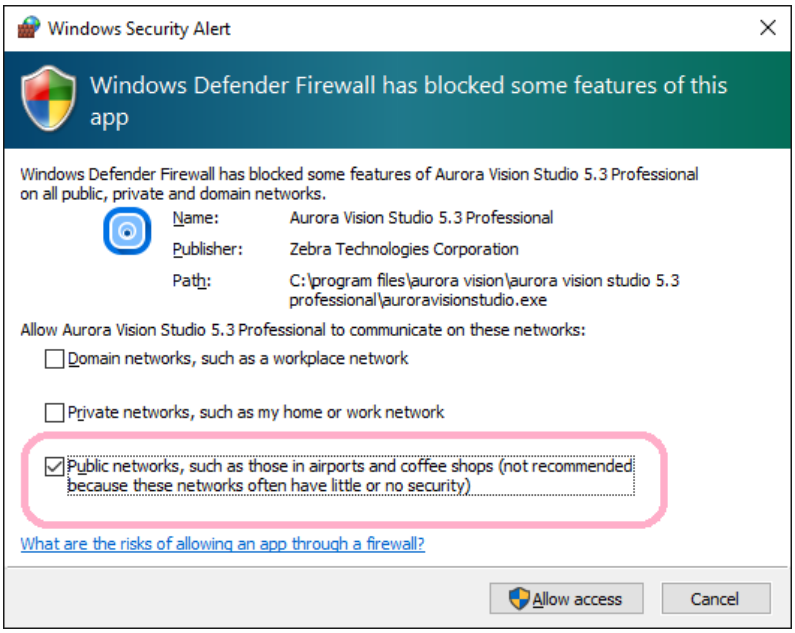
A common case of using multiple cameras at once is to capture multiple photos of a object from a single trigger source (with synchronous triggering):



All above recommendations must be considered for such configuration. Because under synchronous triggering all the cameras will always be transferring images at the same time the problem of momentary burst transfer is especially present. Care must be taken to limit the maximum network transmission speed in the cameras to the system limits and to give enough time between trigger events for the cameras to finish the transfer.

Enabling Traffic in Firewall

Standard Windows Firewall or other active firewall applications should prompt for confirmation to enable incoming traffic upon first start of an Aurora Vision program that is using a video streaming filter. Sample prompt message from standard Windows 7 Firewall is shown in the image below.



Please note that **a device connected directly to the computer's network adapter** in Windows Vista and Windows 7 will become an element of an unidentified network. Such a device will be treated by default as a **Public network**. In order to communicate with such a device, you must allow traffic in Public networks as well, as shown in the image above.

Clicking on *Allow access* will enable the application to stream video from a device. Because of a delay caused by the firewall dialog, the first run of a program may fail with a timeout error. In such situations, just try running the program again after enabling access.

For information about changing settings of your firewall application, search for how to allow a program to communicate through this firewall in Windows Help or a third-party application manual. GigE Vision® driver requires that incoming traffic is enabled on all UDP ports for Aurora Vision Studio and Aurora Vision Executor (by default located in *C:\Program Files\Aurora Vision\Aurora Vision Studio Professional\AuroraVisionStudio.exe* and *C:\Program Files\Aurora Vision\Aurora Vision Studio Runtime\AuroraVisionExecutor.exe*).

Enabling Jumbo Packets

Introduction

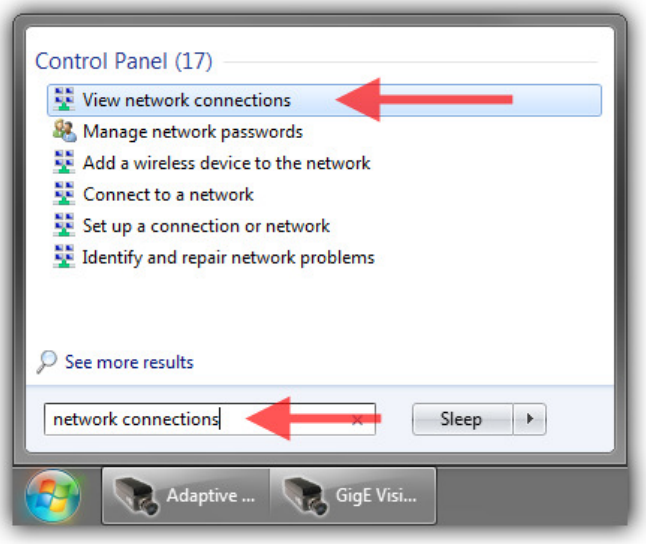
Jumbo Packet is an extension of network devices that allows for transmission of packets bigger than 1.5kB. Enabling Jumbo Packets can significantly increase video streaming performance.

Note that not all network devices support Jumbo Packets. To activate a big packet size, all devices from a network adapter through network routing equipment to a camera device must support and have enabled big packet sizes. Most suitable situation for using Jumbo Packets is when the device is connected directly to computer's network adapter with a crossed Ethernet cable.

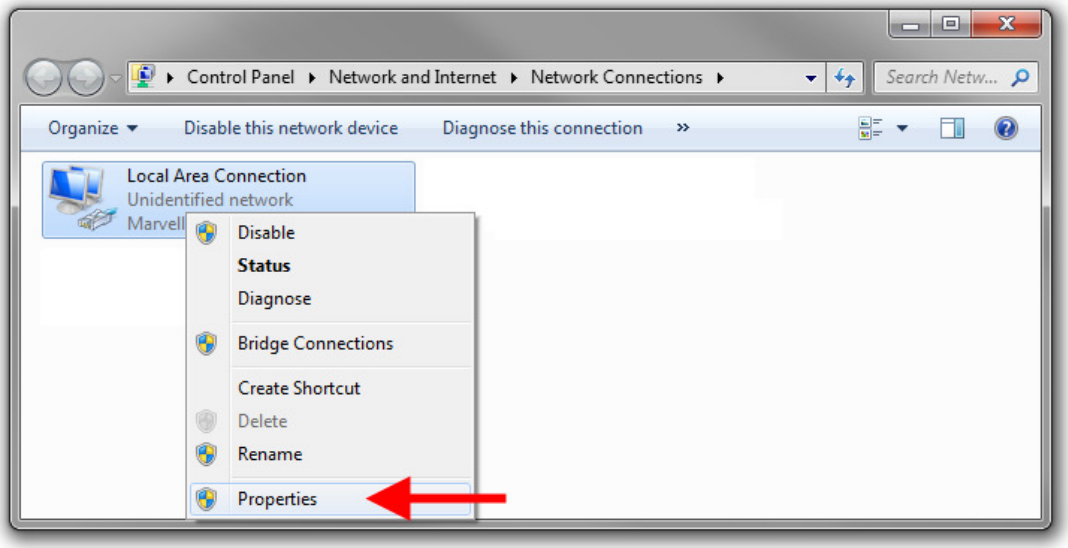
Do not enable Jumbo Packets when the device is connected through complicated network infrastructure with more than one routing path, as maximum allowed packet sizes detected at application start can change later in the process.

Enabling Jumbo Packets in Windows Vista/7

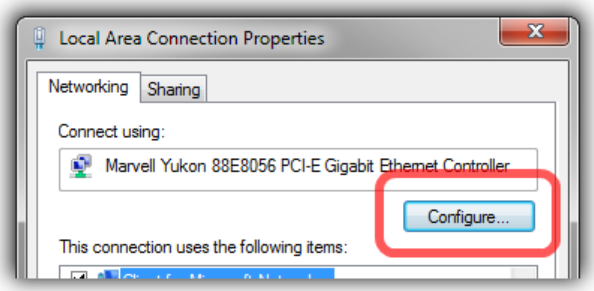
1. Open network connections applet from the control panel.



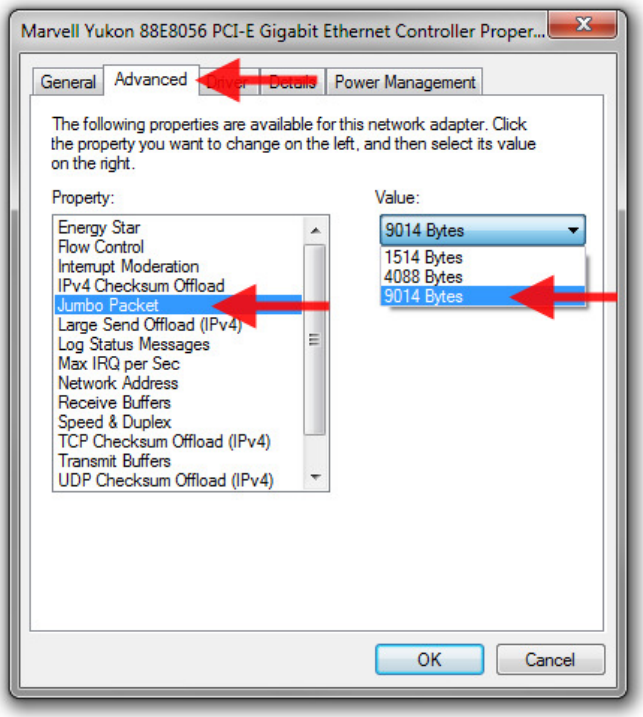
2. Right click a network adapter that have a connection with a device and open its properties (an administrator password may be needed).



3. In network adapter's properties click on *Configure*.



4. From Advanced tab select *Jumbo Packet* property and increase its value up to 9014 Bytes (9k Bytes). This step might look differently depending on the network card vendor. For some vendors this property might have different similar name (e.g. *Large Packet*). When there is no property for enabling/setting large packet size this card does not support jumbo packets.

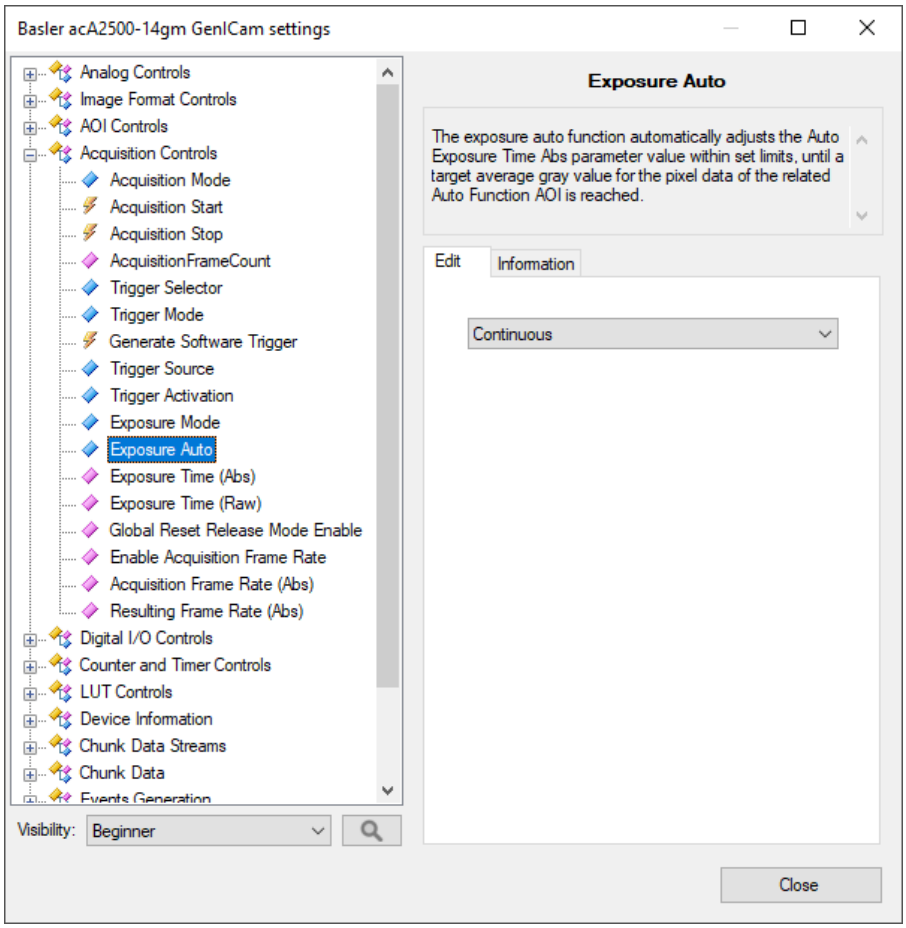


5. Click OK.

Device Settings Editor

GigE Vision® compliant devices implement the GenICam standard that describes camera internal parameters and a way to access them. Device Manager allows a user to access and edit device settings through a Settings Editor tool (available from *Tools » Access Device Settings*).

Example appearance of the Device Settings Editor is shown on the image below.



On the left side of the window is a tree representation of device parameters split into categories. All these parameters and their organization is device dependent, which means that different devices can produce different sets of parameters, with different meanings. Parameter's friendly name and a brief explanation (also provided by a device) is shown on the right side of the window after the parameter is highlighted in tree. For more information about specific parameter functions refer to a device documentation.

When editing selected parameter is possible and supported, an editor of the parameter value will be displayed below its explanation. Different editors are provided for the following parameter types:

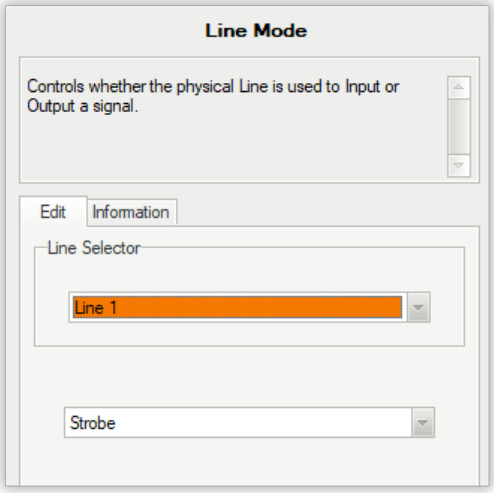
- **Integer** – Plain number or hexadecimal number (indicated by *Hex* label on the left side of the text box). Values are limited by their maximum, minimum and allowed step. Numbers that does not fulfill this rules are corrected automatically upon confirmation. After clicking on *Save* button (or pressing *Enter*) new value will be validated and sent to the device.
- **Float** – Real number with fractional part. Values are limited by their maximum and minimum. Numbers that do not match this range are corrected automatically upon confirmation. A parameter can also have suggested step added after clicking in *+/−* buttons. After clicking on *Save* button (or pressing Enter) new value will be validated and sent to the device.
- **String** – Text of a limited length.
- **Boolean** – Single Yes/No value represented by check box. A value is sent to the device immediately after check state is changed.
- **Enumeration** – Parameter that accepts one of several predefined values. Predefined values are represented as list of their friendly names. Parameter is edited by choosing one of its values from a drop-down list. New values are sent to the device immediately after their selection in the list.
- **Command** – This is a special parameter that is represented only by a single button. Clicking on button will execute related activity in the device (for example Saving current parameter set to non-volatile memory).

Depending on situation, editor can be disabled (grayed), which means that this parameter is currently

locked (for example parameter describing image format when the camera streaming is active). Editor can be read only (Save button grayed, grayed drop-down list or unchangeable check-box), which means that this parameter is read-only (for example informational parameters like manufacturer name).

Instead of an editor, there can also be a text displayed: **"This parameter is currently not available"**. This means that the parameter cannot be accessed or edited in the current device state or due to other parameter states. For example, parameter describing acquisition frame rate value, when the user selection of frame rate is disabled (by parameter like *Enable Acquisition Framerate*).

Sometimes with the parameter editor displayed will be an additional editor, named selector.



In such a situation, the selected parameter is connected with one of categories (slots) described by the selector. In the example in the image above, the parameter determines whether the physical line is used to input or output a signal. This device has two lines and both have their own separate values to choose from. The selector picks which line we want to edit, and the bottom editor changes its purpose. This means that there are actually two different *Line Modes* parameters in the device.

Please note that selector will not always be displayed above editor. You must follow a device documentation and search parameters tree for selectors and other parameters on which this parameter is dependent.

The Device Settings Editor can be used to identify device capabilities and descriptions or to set up a new device. Device Editor can be also used when a program is running and the camera is streaming. In this situation changes should be immediately visible in the camera output.

Settings Editor gives a user unlimited access to the device parameters and, **when used improperly, can put the device in an invalid state** in which the device will become inaccessible by applications or can cause transitional errors in the program execution.

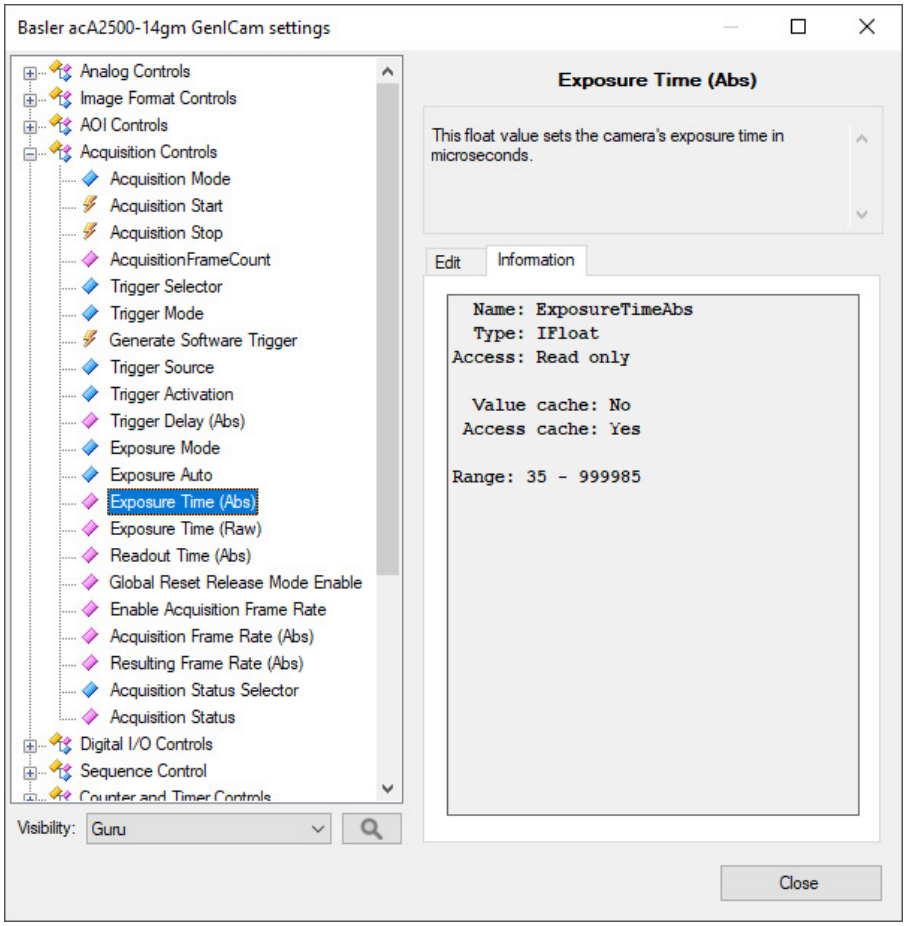
Saving Device Configuration

Most parameters available in the Settings Editor are stored by devices in a volatile memory and will be lost (reset to default) after device reset or power down.

A device should offer functions to save parameters set in *Configuration Sets* section of parameters tree. Refer to a device documentation for more information about configuration set saving and loading.

Selecting Parameter Name for Filter

The Settings Editor can also be used to select parameter names for filters in Aurora Vision Studio. You can access this mode by clicking the *...* button in the filter properties list. When there is more than one device in the network, one must be first selected in the manager. The tool will show parameters of the specified device for its current state.



In this mode the window appearance is changed. Instead of editors, useful information for a program developer is displayed on the right side of the window, including:

- **Name** – parameter internal name. Note that parameter tree and descriptions are using human friendly name, not parameter ID. This field shows a proper parameter ID that must be used in a filter property. You can easily transfer this name to property list by clicking Select button (or double clicking parameter in tree).
- **Type** – parameter type name. This type must be consistent with the filter value type.
- **Access** – allowed access to parameter. Parameter must be writable to be set by program.
- **Range** – for numeric parameters this field shows the allowed range. Range of some parameters can change dynamically during its operation.
- **Value cache** – when GenAPI cache is enabled this field indicates if device allows to store this parameter value in local memory to reduce network operations.
- **Access cache** – when GenAPI cache is enabled this field indicates if device allows to store access mode of this parameter in local memory to reduce network operations on controlling parameter accessibility.
- **Available entries** – for enumeration parameters this field will list currently available values for a parameter. The field shows proper internal IDs that should be used when setting the parameter (note that editor's drop-down lists are using human friendly names).

Known Issues

In this section you will find solutions to known issues that we have came across while testing communication between Aurora Vision products and different camera models through GigE Vision.

Table of Contents

1. The Imaging Source cameras
2. Flir cameras
3. Pixel Formats for 3D Data
4. Basic Troubleshooting

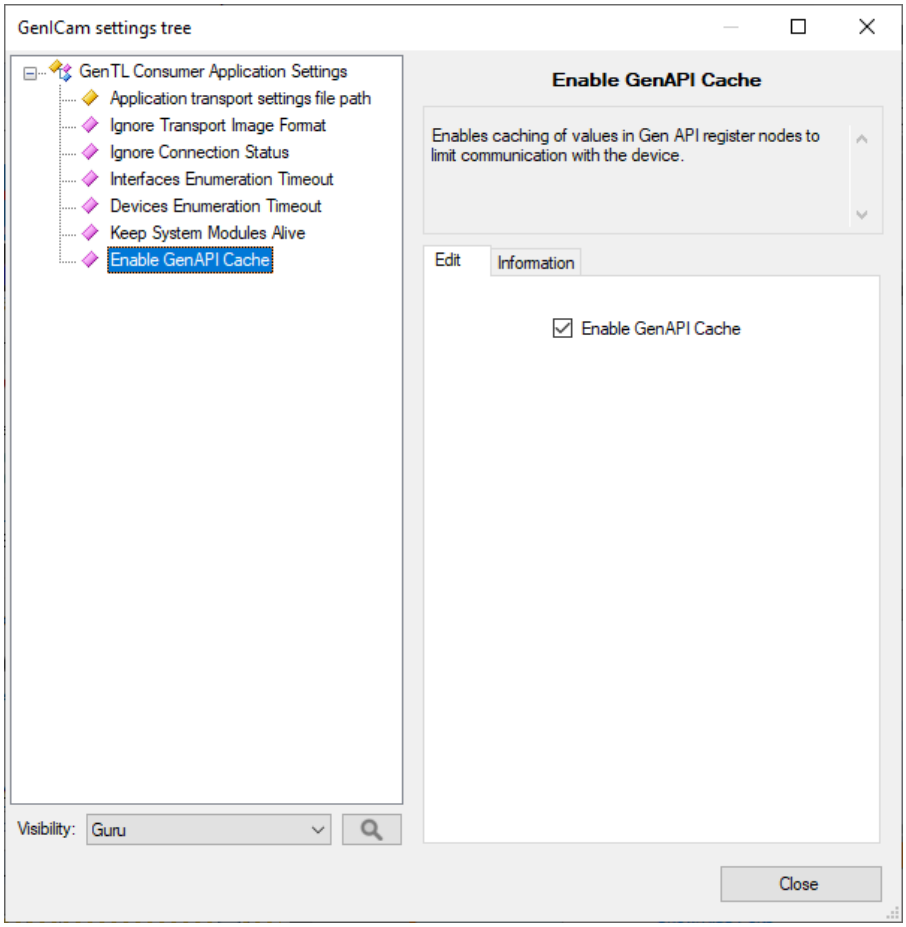
The Imaging Source cameras

There might be problems with image acquisition from The Imaging Source cameras through GigE. It's caused by the implementation (regarding caching and packet size) of GigE Vision standard in those cameras and as a result no image can be seen in Aurora Vision Studio (the previews are empty during program execution).

To resolve this issue, a camera restart (this has to be done only once, after you encounter the problem with image acquisition) and changing some parameters in Aurora Vision GenAPI configuration are required. Parameters which should be changed are:

- Enable *GenAPI Cache* (should be set to *False*),
- Disable *Packet Size Negotiation* (should be set to *True*),
- Enable *Constant Packet Size* (should be set to *False*).

You can change these parameters in the GigEVision Application Settings Tree (*Tool » Manage GigE Vision Devices... » Tools » Application Transport Settings*). Make sure that each of the three parameters mentioned above is set to the proper value in this tree.



Firmware issues

Please note that some camera models, such as **DFK 39GR0522-Z12** with certain firmware versions (e.g., **Rev 3235, FPGA GigE3LZ-HISP:5**), do not work properly with GigE Vision tools. If you encounter such obstacles, you can apply one of the following solutions:

- Contact The Imaging Source support at support@theimagingsource.com to get a newer firmware version (e.g., tested **Rev 3275, FPGA GigE3LZ-HISP:5** works). Follow the firmware update instructions.
- Alternatively, download GenTL Producer for GigE Vision Cameras and IC Imaging Control 4 SDK, and use GenICam or The Imaging Source tools.

Flir cameras

There might be problems with image acquisition from Flir cameras through GigE Vision. It is caused by the implementation (regarding image trailer data) of the GigE Vision standard in those cameras and as a result no image can be seen in Aurora Vision Studio (the previews are empty during program execution).

To resolve this issue, a parameter change in Aurora Vision GenAPI configuration is required. Parameter which should be changed is:

- Enable *Ignore Gev Image Trailer data* (should be set to *True*).

You can change these parameters in the GigE Vision Application Settings Tree (Tool » Manage GigE Vision Devices... » Tools » Application Transport Settings). Remember to revert this changes before working with different devices as it can prevent some cameras from working properly in some modes.

Pixel Formats for 3D Data

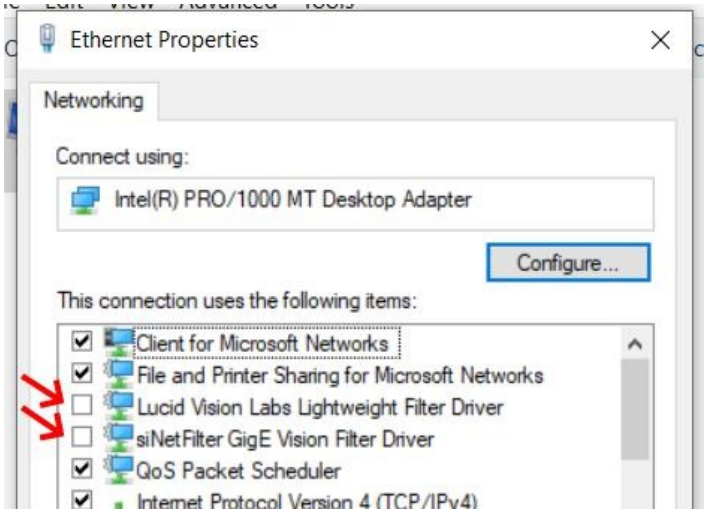
We have noticed that when setting a pixel format for 3D data in the camera (e.g., Coord3D_ABC8, Coord3D_ABC32f, etc.), specifying the same Pixel Format in the GigEVision_GrabImage filter often results in an error: **Received stream payload does not contain an image or the image format is invalid.**

In such cases, it is better to set the pixel format in the Device Manager or by using GigEVision_SetParameter filters, and leave the **PixelFormat** input in the GigEVision_GrabImage filter empty (unselected).

Basic Troubleshooting

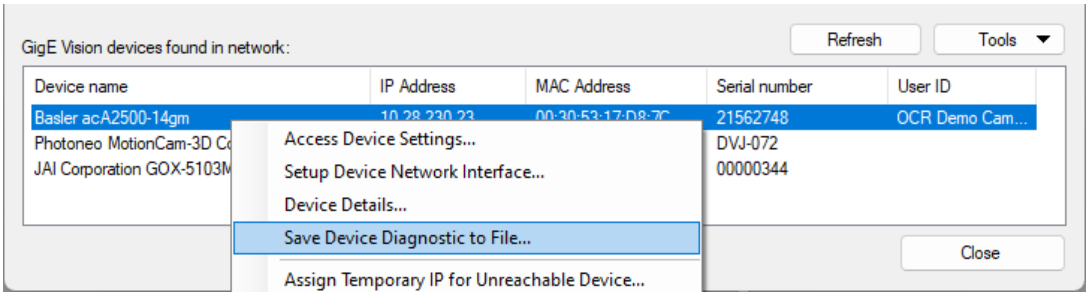
If a GigEVision device is connected, but the image cannot be acquired, please follow the steps below.

1. Check if the GigEVision_GrabImage tool is in the **ACQUIRE** section. If not, move it to the right section.
2. If the camera settings have been changed, set the default value for each changed parameter.
3. Check if there is no other software that uses the camera in the background. If there is, close it.
4. Check the camera connection. Connect the device directly to the PC if possible. If not, simplify the connection as much as possible. If there is any managed switch, check its settings.
5. If possible, use stable power supply (not PoE).
6. Set static IP address for both the camera and the PC. More information can be found here.
7. Enable Traffic in Firewall. More information can be found here.
8. Turn off Third-party Network Card Drivers.



9. Enable Jumbo Packets. More information can be found here.
10. Update PC Network Card Driver.

If after following all the steps the issue still appears, open GigE Vision Device Manager, right-click on the issued camera, choose ***Save Device Diagnostic to File***, and send the created file to the Technical Support Team.



Zebra Aurora™ Vision

This article is valid for version 5.7.5

Legal Terms of Use Privacy Policy

ZEBRA and the stylized Zebra head are trademarks of Zebra Technologies Corp., registered in many jurisdictions worldwide. All other trademarks are the property of their respective owners. ©2026 Zebra Technologies Corp. and/or its affiliates.