



Gocator Acquisition for Cognex VisionPro

Overview	1
Background	1
Environment and Prerequisites	2
Usage	4
Method 1 - Using VisionPro QuickBuild	4
Method 2 - Programming with VisionPro .NET Libraries	8

Revision History

Revision	Changes	Date	Author
1.0	Initial draft	6/1/2017	ocohen

Overview

VisionPro is a powerful vision tool, we would like to allow our customers to use it with the Gocator directly.

Background

Developing a VisionPro application can be done in 4 ways (taken from the VisionPro Help utility):

- Path 1:
Interactive development with QuickBuild plus the Application Wizard-Generated Operator Interface
- Path 2:
Interactive development with QuickBuild plus a customized Application Wizard Operator Interface
- Path 3:
Interactive development with QuickBuild plus a Custom Operator Interface



- Path 4:
Custom application using native VisionPro .NET Programming Interface

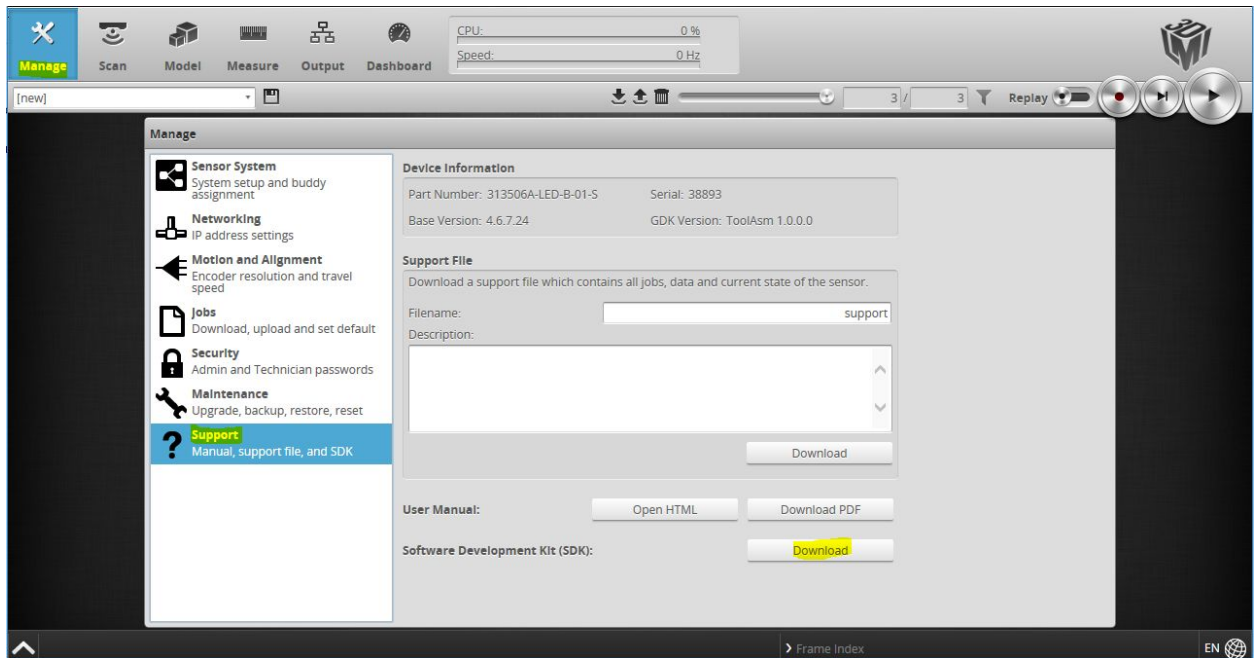
Specifically when trying to control the acquisition stage, the division is to two methods:

- [Method 1](#) (includes Paths 1-3):
Creating the application using VisionPro QuickBuild.
- [Method 2](#) (includes Path 4):
Programming a custom application using the VisionPro .NET libraries.

In this document I will describe how to acquire an image from the Gocator as input to your VisionPro application.

Environment and Prerequisites

- Cognex VisionPro 9.0
Vision Pro 9 can be downloaded [here](#), and must be used with a Cognex dongle.
- Gocator SDK 4.6.7.17
Available on LMI's website. A link can be found in the Gocator GUI:



Refer to the Gocator SDK help for more information on installing and using it.

- The Gocator must be set up in either:
 - Surface Mode - Surfaces must be set to output via Gocator protocol. Each surface is handled as a separate image.



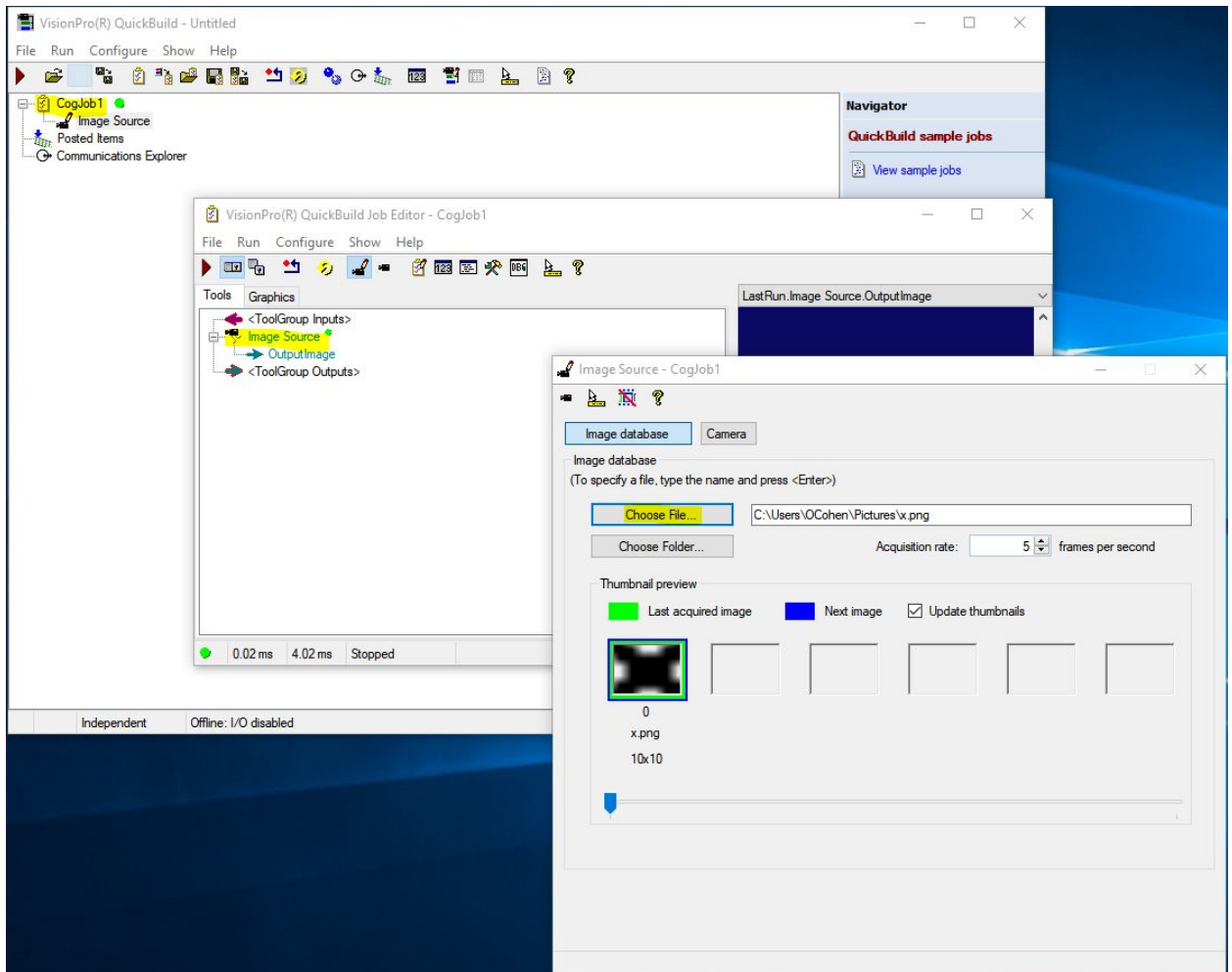
- Profile Mode - Profiles must be set to output via Gocator protocol. Profiles are treated as individual images with 1 pixel height.

Usage

Method 1 - Using VisionPro QuickBuild

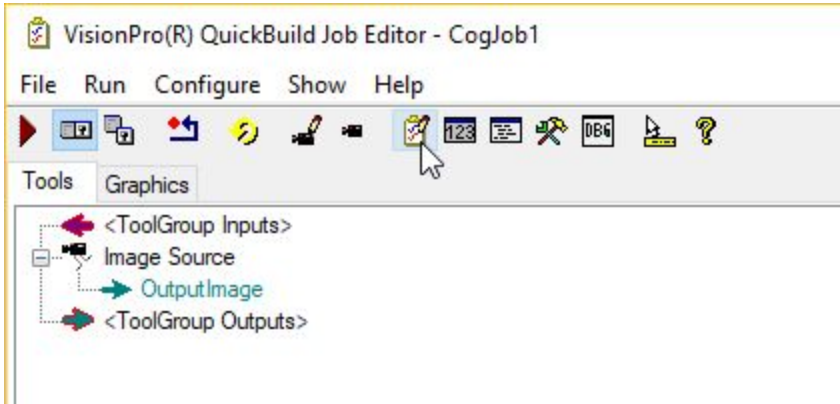
In this method we will need to load a dummy image file, then use an SDK script to overwrite it with the Gocator's output image.

- 1) Create a QuickBuild Project and set the image source to an image file on your drive. This file can be any image file of any format and size.

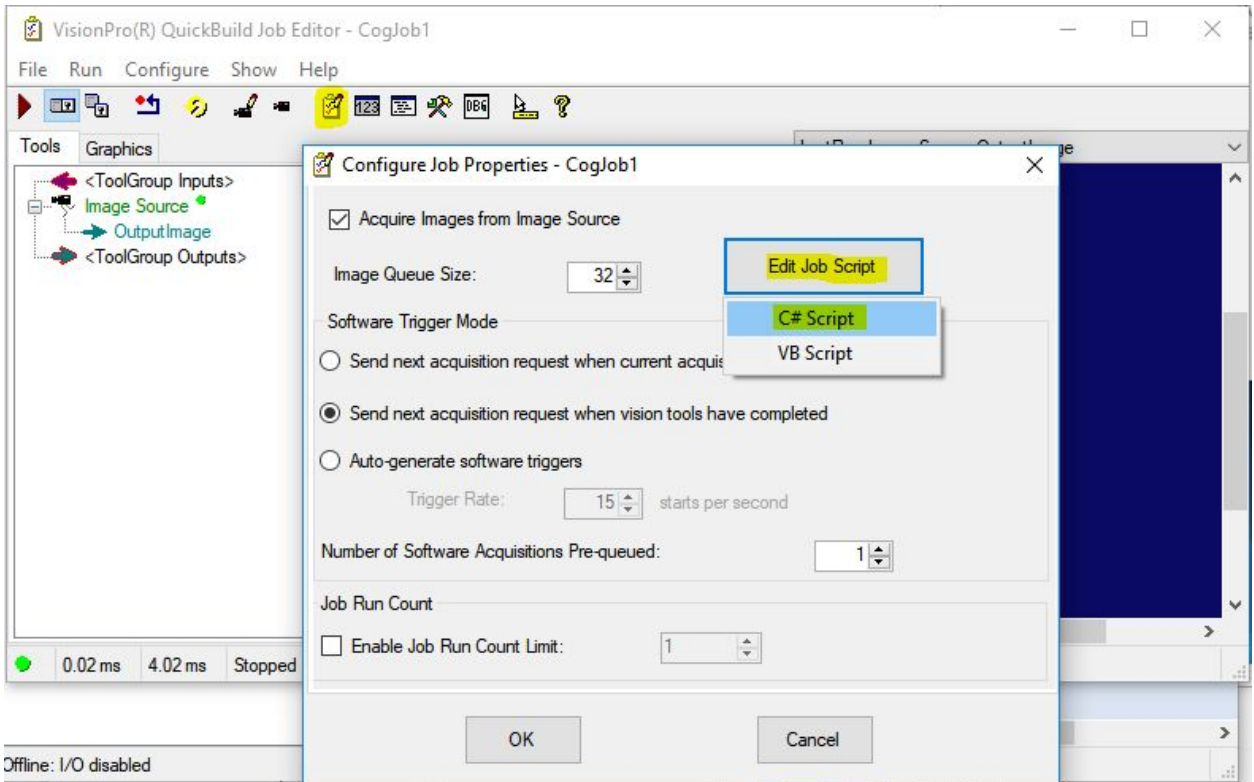




- 2) Open the “ Job Properties” in the Job Editor window



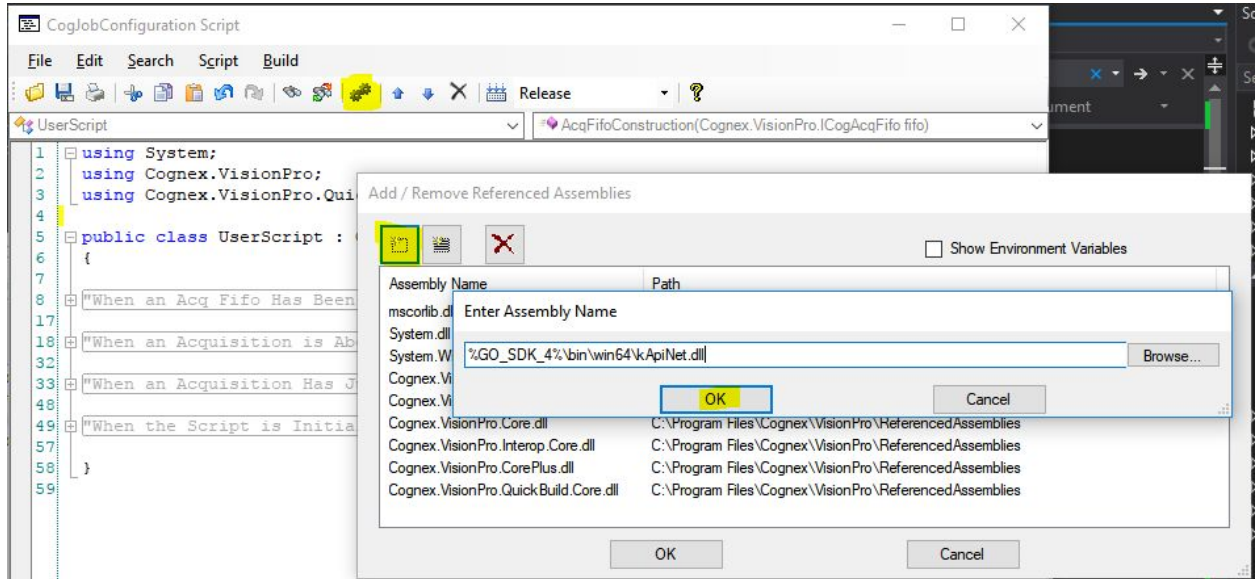
- 3) Select “Edit Job Script” and select C# from the menu (VB Script is also available if needed)




- 4) Click on “ Add/Remove References”
- 5) Click “ Add Reference by typing the name...” and enter the path to these two files:
%GO_SDK_4%\bin\win64\kApiNet.dll
%GO_SDK_4%\bin\win64\GoSdkNet.dll



And Click “OK”



- 6) Click the “ Open” button and load the provided script file:
https://ftp.lmi3d.com/main.html?download&weblink=b55d5d3fb6cf3480c92aba6e7811b423&realfilename=CogJobConfiguration_script.cs
Download password: gocator

If you intend to use VisionPro’s script abilities for your own pre/post processing of the image, you may do so by altering the provided code.

- 7) In the “Initialize” function, change the constant SENSOR_IP to your own Gocator’s IP address. The address “127.0.0.1” is used to acquire from a GoEmulator.

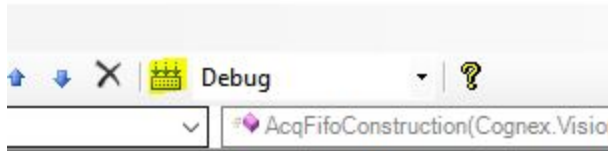
```
const string SENSOR_IP = "127.0.0.1"; // Change this to your Gocator's IP address.
```

- 8) In the “PostAcquisitionRefInfo” function, you will see two images are returned from the Gocator, a height map and an intensity image. You can choose which of them you want to use.

```
image = heightMap; // replace this with "intensity" to get intensity data
```



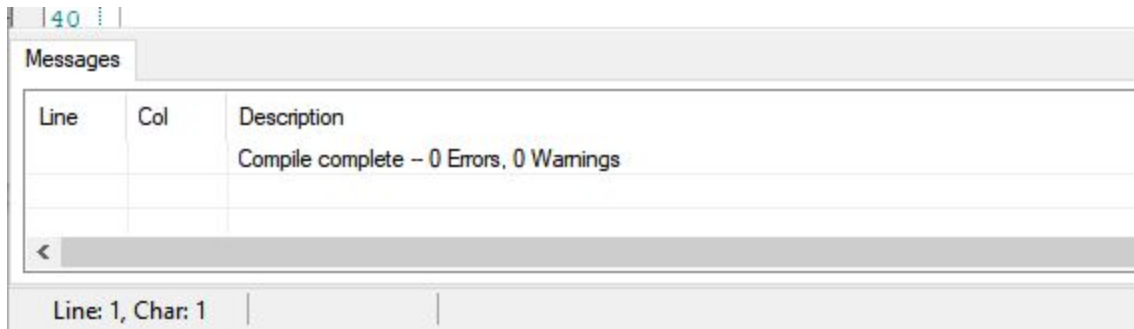
9) Build the script by clicking on the “Build” button



```
services;
```

Note that currently the Gocator SDK integration only works when building in “Debug” mode.

10) Make sure you have no build errors or warnings



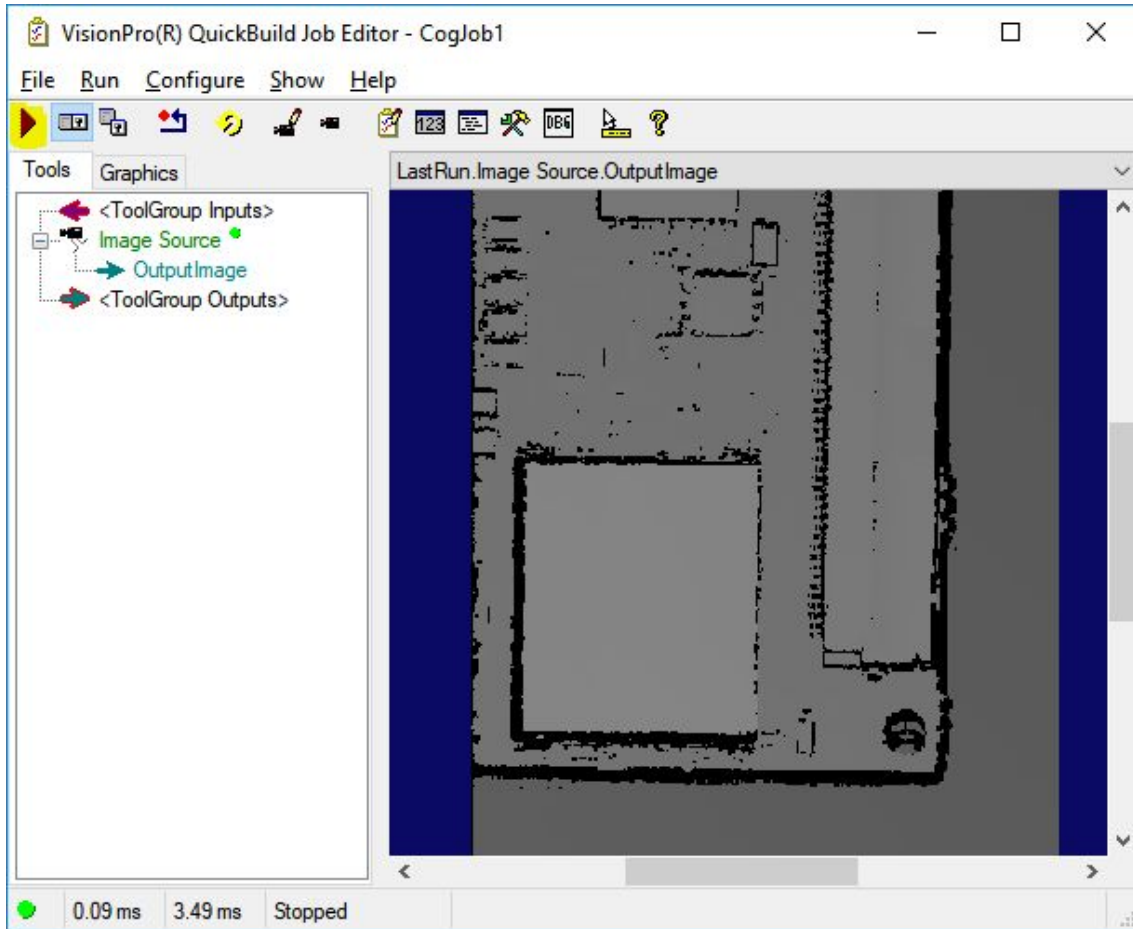
11) Close the script window and **Press OK** in the “Job Properties” window

If you do not press OK, your changes to the script will not be saved

12) At this point a connection to the Gocator will be made.



13) Press “▶ Run Job Once” to acquire a single image, or “🔄 Run Job Continuously” to constantly receive images.





Method 2 - Programming with VisionPro .NET Libraries

In this method you will use the Gocator's SDK to connect to a sensor and acquire an image. Then the image must be converted to the ICogImage format for further processing using the VisionPro libraries.

- 1) Add reference to the SDK DLLs to your .NET projects:
 %GO_SDK_4%\bin\win64\kApiNet.dll
 %GO_SDK_4%\bin\win64\GoSdkNet.dll
- 2) Look at the provided project for a C# example on how to acquire and convert an image from the Gocator.

See the SDK help for more information.