

Vision Toolbox for MATLAB

Release Notes

**Embedded Target for the S32V234 Family of Automotive Vision Processors
Version 1.2.0**

Summary

| | | |
|-----|--|------|
| 1 | What is New | 1-3 |
| 1.1 | NXP Vision Toolbox version 1.1.0 | 1-3 |
| 2 | Vision Toolbox Features | 2-6 |
| 2.1 | Packages & Derivatives | 2-6 |
| 2.2 | Code Generation | 2-6 |
| 2.3 | Functions and Kernels | 2-6 |
| 2.4 | S32V234 Example Library | 2-7 |
| 3 | Prerequisites | 3-8 |
| 3.1 | OSes Supported | 3-8 |
| 3.2 | MATLAB Required and Recommended Products | 3-8 |
| 3.3 | Build Tools | 3-9 |
| 3.4 | Vision SDK package..... | 3-10 |
| 4 | Vision Toolbox Extras..... | 4-11 |
| 4.1 | Evaluation Board Support..... | 4-11 |
| 4.2 | Target Connection | 4-11 |
| 5 | Known Limitations..... | 5-12 |
| 6 | Support Information | 6-12 |

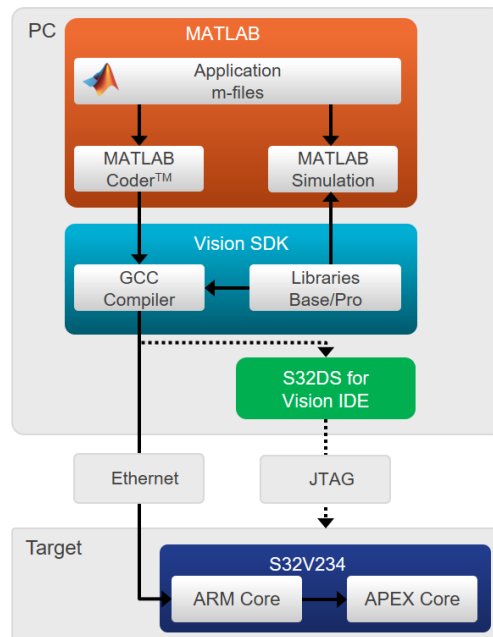
1 What is New

1.1 NXP Vision Toolbox version 1.1.0

This release of [NPX Vision Toolbox version 1.2.0](#) for MATLAB is designed to support the S32V234 Automotive Vision Processors for automatic code generation and NXP target deployment of vision applications for S32V234 directly from MATLAB environment.

The NXP Vision Toolbox allows the following software development flows:

- Implementation and simulation of Computer Vision applications in MATLAB environment using MATLAB built-in Vision Classes or NXP Vision Toolbox MEX wrappers for NXP Vision SDK classes
- Implementation and simulation of Machine Learning applications in MATLAB environment using Deep Learning Toolbox
- Automatic code generation from MATLAB m-scripts to C++ code
- Application deployment on NXP S32V234 evaluation boards (SBC and EVB) via Ethernet port directly from MATLAB environment.
- Application debugging via S32 Design Studio for Vision



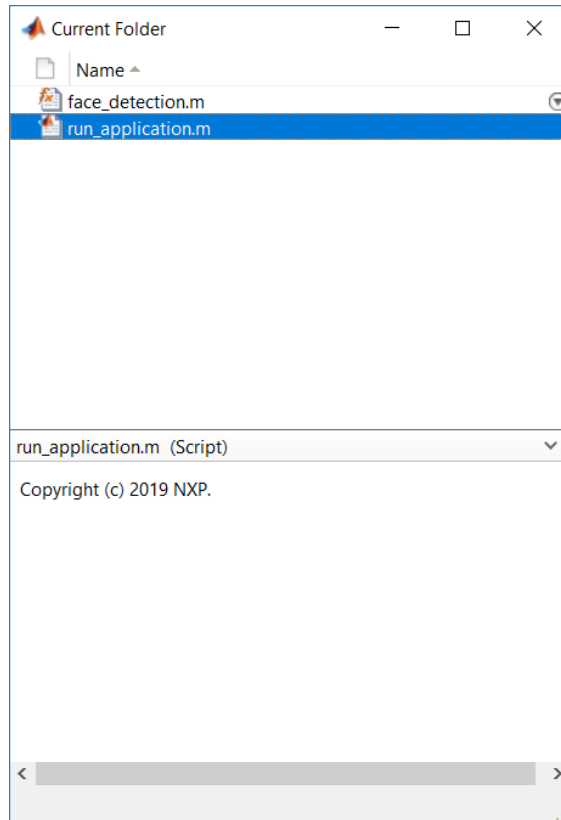
For more information please check the Vision Toolbox quick start guide or watch [this](#) webinar.

[NXP Vision Toolbox version 1.2.0](#) new features compared with previous release (1.1.0):

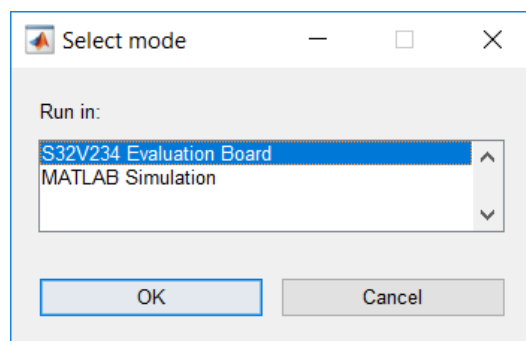
- Latest Vision SDK version 1.4.0;
- Latest MATLAB R2019b release;
- New way to pass parameters to the entry function at codegen time;
- New Wizard for running/deploying examples on the target S32V234 board;

- New CNN example for traffic sign detection made with Transfer Learning on SqueezeNet with the Belgian Traffic Signs Dataset.

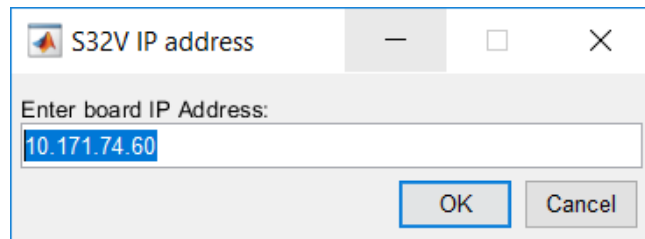
Inside most examples folders from the NXP Vision Toolbox you can find a `run_application.m` script which is a Wizard (a GUI) that lets you simply run the examples in both simulation and on the target.



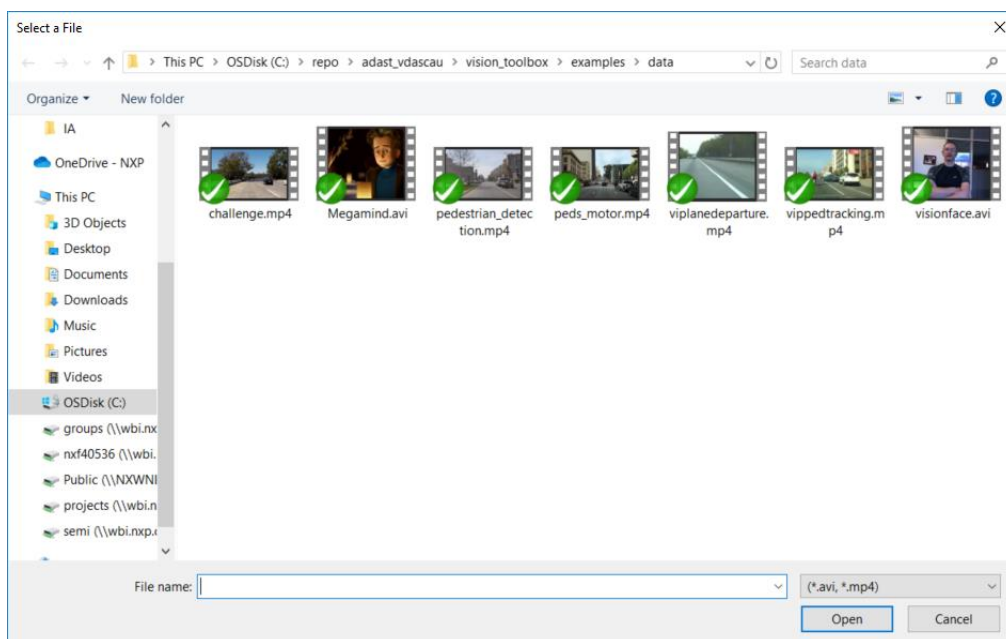
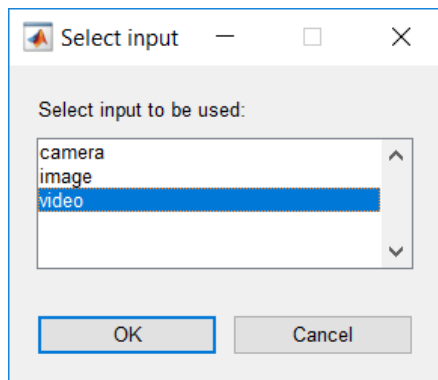
Execute the `run_application.m` script. In the first step you can choose if you want to run the example in Simulation or on the S32V234 board.



When choosing the S32 board, the Wizard will prompt you for the IP Address:



After that you will be asked to select the input source. In case you select a video or an image you can then browse your computer for the respective file and select it.



This works in the exact same way (obviously without the IP address) for simulation. In case you've selected the camera as the input option, you will also be asked for the MIPI port to be used (MIPI-A, MIPI-B). After this step it will start compiling and deploying the example to the board with the selected input source and preferences.

For more details please refer to the following chapters.

2 Vision Toolbox Features

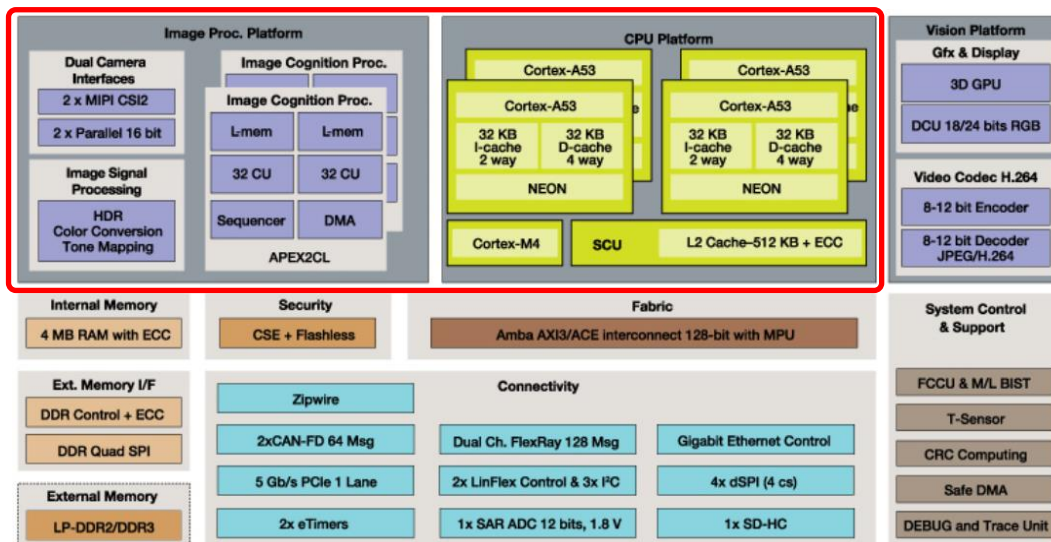
2.1 Packages & Derivatives

The NXP Vision Toolbox version 1.2.0 supports:

- NXP [S32V234](#) Vision Processor.
- Testing and validation has been completed on production qualified parts mounted on [S32V234 Evaluation Board](#) and [SBC-S32V234 Evaluation Board](#) equipped with a [S32V-SonyCam](#)

2.2 Code Generation

NXP Vision Toolbox support code generation for the devices and functions highlighted on top of the S32V234 block diagram below:



2.3 Functions and Kernels

The NXP Vision Toolbox for S32V234 kernels organization follows the Vision SDK implementation and supports several component libraries:

- Arithmetic kernels - provide basic operators for element-wise addition, subtraction, multiplication, division and arithmetic shifting
- Comparison kernels - provide basic element-wise comparison operators like less than, less-than-or-equal, binary AND operator and binary descriptor matches
- Conversion kernels - support conversion from 16 to 8 bit and from RGB format to grayscale
- Display kernels - provide examples of marking an image at certain points as overlay or in a certain color channel.
- Feature detection - provides two corner detection algorithms FAST9 and Harris corner detection

- Filtering - offers kernels for general purpose filtering, and also the most used filters like Gaussian filtering, gradient computation, non-maximum suppression and saturation
- Geometry - provides geometric transformations, like rotations and bilinear interpolation and also a replacement for indirect inputs, called offset selection
- Morphology - example of a morphological dilation operator.
- Object detection - two object detection algorithms: Haar cascade and LBP (local binary pattern) cascade
- Optimization - implementation of the Integral Image (SAT) kernel and a SAT-based box filter.
- Resizing - provides downsampling and upsampling kernels (gives examples of size changes inside a filter)
- Statistics - provides kernels for statistics computations, such as a Histogram kernel, a vector-to-scalar reduction kernel and an accumulation kernel.

To use any of these kernels in the MATLAB m-script functions use:

```
nxpvt.apu.<kernel_name>(args1, ...)
```

2.4 S32V234 Example Library

S32V234 Examples Library represents a collection of MATLAB models that let you test and run complex applications. These examples are grouped in different layers that mimics a typical development flow.

The main functionalities supported are:

- Vision Applications;
- CNN Applications;
- APEX Kernels;
- APEX Computer Vision;
- S32V234 IO Examples;

The MATLAB models shown as examples are enhanced with a comprehensive description to help users understand better the functionality that is exercised, hardware setup instructions whenever are necessary and a result validation section.

3 Prerequisites

3.1 OSES Supported

The NXP Vision Toolbox is supported only on PC with Windows OS. For a flowless development experience the minimum recommended PC platform is:

- Windows® 7/10 64bit Operating System
- At least 2 GHz CPU Speed
- At least 4 GB of RAM
- At least 20 GB of free disk space.
- Internet connectivity for web downloads.

| Operating System Supported | SP Level | 64-bit |
|----------------------------|----------|--------|
| Windows 7 | SP1 | X |
| Windows 10 | | X |

3.2 MATLAB Required and Recommended Products

The NXP Vision Toolbox for S32V234 requires the following MathWorks products to be installed. Make sure you have a valid license for the products marked as “Required”

| Product | Version Compatibility | Required or Recommended |
|--|-----------------------|---|
| MATLAB | R2018a or newer | Required |
| MATLAB Coder | R2018a or newer | Required |
| Embedded Coder | R2018a or newer | Required |
| Image Processing Toolbox | R2018a or newer | Required |
| Computer Vision System Toolbox | R2018a or newer | Required |
| Embedded Coder Support Package for ARM Cortex-A Processors | R2018a or newer | Required |
| Computer Vision System Toolbox OpenCV Interface | R2018a or newer | Required |
| Deep Learning Toolbox | R2018a or newer | Required for CNN code generation |
| MATLAB Support Package for USB Webcams | R2018a or newer | Recommended for webcam usage in simulations |

| | | |
|--|-----------------|---|
| Image Acquisition Toolbox Support Package for OS Generic Video Interface | R2018a or newer | Recommended for webcam usage in simulations |
| Deep Learning Toolbox™ Model for SqueezeNet Network | R2018a or newer | Recommended for CNN code generation |
| Deep Learning Toolbox™ Model for AlexNet Network | R2018a or newer | Recommended for CNN code generation |
| Deep Learning Toolbox™ Model for GoogLeNet Network | R2018a or newer | Recommended for CNN code generation |
| MATLAB Coder Interface for Deep Learning Libraries Deep Learning Toolbox | R2018a or newer | Recommended for CNN code generation |

Due to code generation performance issues the NXP Vision Toolbox uses a special feature `row-major` that has been introduced in [MATLAB Coder 2018a](#). This feature allows better code generation that is compatible with embedded systems designed to store the arrays in row-major format avoiding this way unnecessary copies or transposes between MATLAB and Vision SDK APIs.

3.3 Build Tools

The NXP Vision Toolbox supports code generation for the NXP ARM GNU and NXP APU compilers.

| Compiler | Versions Tested |
|----------------------|-----------------|
| NXP ARM GNU Compiler | NXP GCC 6.3.1 |
| NXP APU Compiler | V1.0 build 530 |

The target compilers used for NXP Vision Toolbox needs to be configured. Use the notation below to setup these compiler environment or user variables. Ensure that such variables are defined to compiler path value as shown below:

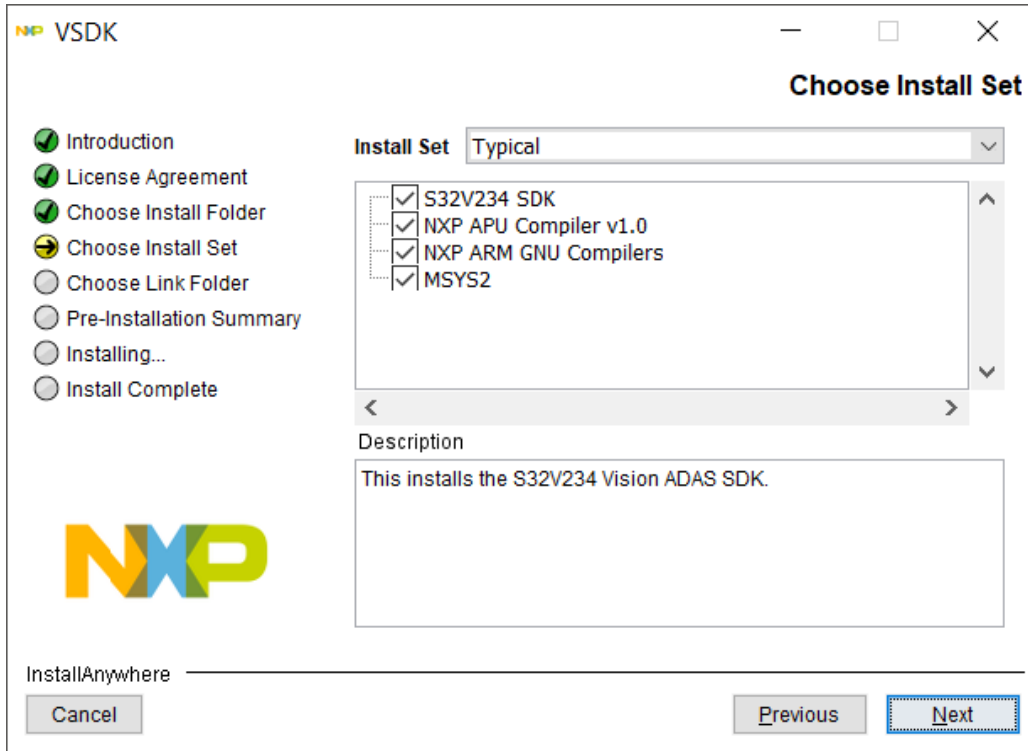
```

APU_TOOLS= C:/NXP/APU_Compiler_v1.0
S32V234_SDK_ROOT = C:/NXP/VisionSDK_S32V2_RTM_1_4_0/s32v234_sdk

```

3.4 Vision SDK package

All the code generated by NXP Vision Toolbox is based on [S32V234 Vision SDK 1.4.0 package](#). This software package is also free of charge and apart of optimized kernels and libraries for the S32V automotive vision processors, it also contains the build tools to cross-compile the MATLAB generated code to ARM A53 and APEX cores. All Vision SDK components must be installed prior of using the NXP Vision Toolbox for S32V234



NOTE If you have installed previous versions of the Vision SDKs, make sure the paths are set correctly to point towards version 1.4.0. The NXP Vision Toolbox version 1.2.0 works in sync with specific VSDK versions.

4 Vision Toolbox Extras

NXP's Vision Toolbox enables additional functionalities that are not MCU specific but could help faster prototyping, validation and verification of the developed models.

4.1 Evaluation Board Support

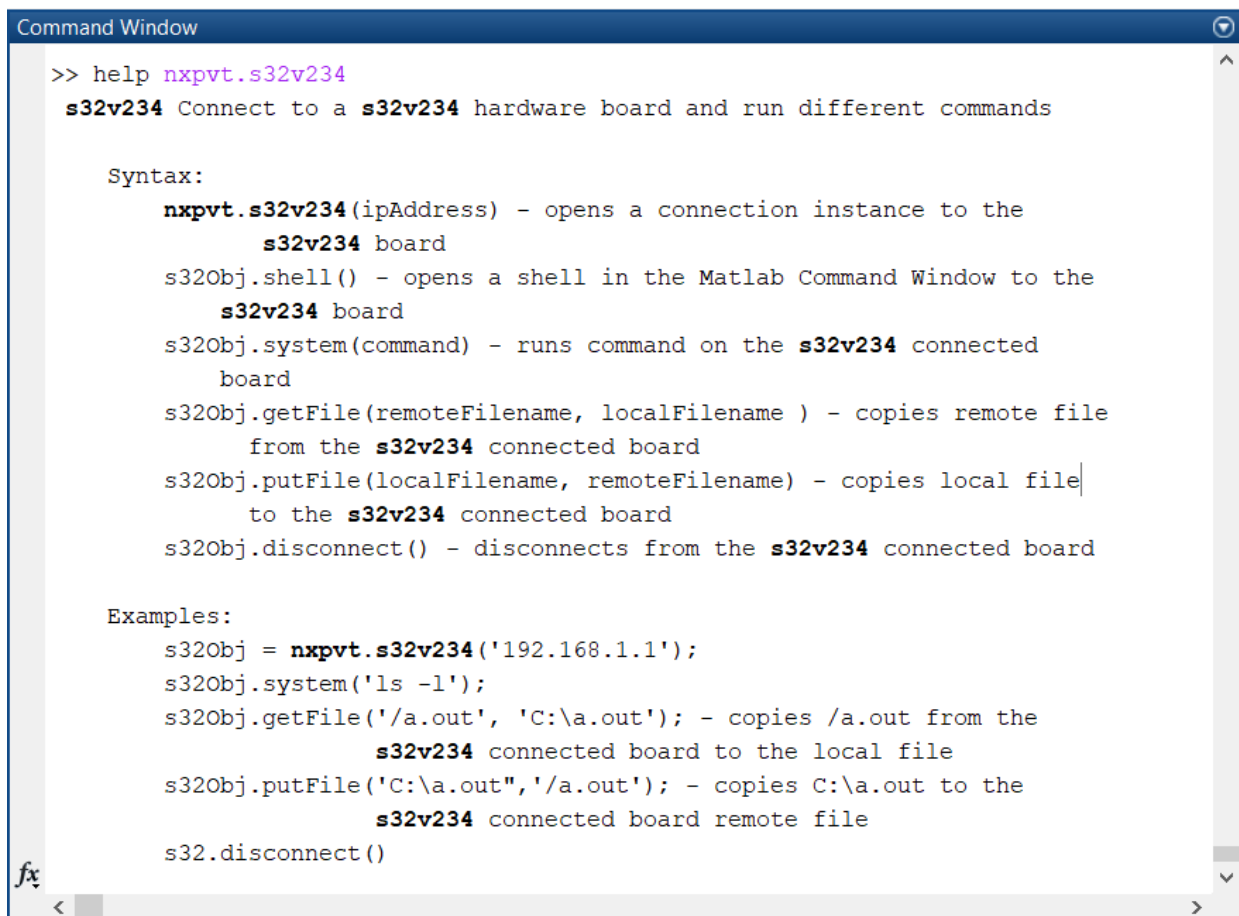
The NXP Vision Toolbox has built in capabilities to configure the SD-Card directly from Windows OS. A Linux OS configured to boot up the S32V234 platform to enable the Eth, ISP, ACF and other various drivers and modules is needed for running embedded vision algorithms.

NXP Vision Toolbox facilitates:

- Easy SD-Card configuration directly from Windows OS via pre-build bootable image
- `nxpvt_create_target()` function for configuration directly from MATLAB

4.2 Target Connection

The NXP Vision Toolbox has built in capabilities to connect and configure the S32V234 microprocessor and NXP evaluation boards. Use `nxpvt.s32v234` object to download and control the target. The list of supported functionalities is shown below:



```
Command Window
>> help nxpvt.s32v234
s32v234 Connect to a s32v234 hardware board and run different commands

Syntax:
  nxpvt.s32v234(ipAddress) - opens a connection instance to the
    s32v234 board
  s32obj.shell() - opens a shell in the Matlab Command Window to the
    s32v234 board
  s32obj.system(command) - runs command on the s32v234 connected
    board
  s32obj.getFile(remoteFilename, localFilename) - copies remote file
    from the s32v234 connected board
  s32obj.putFile(localFilename, remoteFilename) - copies local file
    to the s32v234 connected board
  s32obj.disconnect() - disconnects from the s32v234 connected board

Examples:
  s32obj = nxpvt.s32v234('192.168.1.1');
  s32obj.system('ls -l');
  s32obj.getFile('/a.out', 'C:\a.out'); - copies /a.out from the
    s32v234 connected board to the local file
  s32obj.putFile('C:\a.out', '/a.out'); - copies C:\a.out to the
    s32v234 connected board remote file
  s32.disconnect()
```

5 Known Limitations

This release contains some limitations in terms of code generations:

- 4 classes from VSDK 1.4.0 BASE are not supported for Code Generation nor MEX Simulation: `ColorConverterHT`, `ConvolveFilterHT`, `DerivativeXFilterHT`, `SobelXFilterHT`
- Limited support for VSDK 1.4.0 PRO classes: only `Resize` and `Remap` are supported for code generation and MEX simulation
- Avoid any paths that includes spaces. Please install your MATLAB and VisionSDK related software into short paths without spaces.

6 Support Information

To order NXP products, consult your local sales representative.

For technical support please sign on to the following NXP's Model-Based Design Toolbox Community: <https://community.nxp.com/community/mbdt>

How to Reach Us:

Home Page:
www.nxp.com

Web Support:
www.nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

NXP Semiconductor reserves the right to make changes without further notice to any products herein. NXP Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. NXP Semiconductor does not convey any license under its patent rights nor the rights of others. NXP Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the NXP Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use NXP Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold NXP Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that NXP Semiconductor was negligent regarding the design or manufacture of the part.

MATLAB, Simulink, Stateflow, Handle Graphics, and Real-Time Workshop are registered trademarks, and TargetBox is a trademark of The MathWorks, Inc. Microsoft and .NET Framework are trademarks of Microsoft Corporation. Flexera Software, FlexIm, and FlexNet Publisher are registered trademarks or trademarks of Flexera Software, Inc. and/or InstallShield Co. Inc. in the United States of America and/or other countries. NXP, the NXP logo, CodeWarrior and ColdFire are trademarks of NXP Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. Flexis and Processor Expert are trademarks of NXP Semiconductor, Inc. All other product or service names are the property of their respective owners

©2019 NXP Semiconductors. All rights reserved.

