e-con Systems India Pvt Ltd

RISHABH INFO PARK,
RR Tower – IV, 7th Floor,
Super A-16 * A-17, Thiru-Vi-Ka Industrial Estate,
Guindy, Chennai - 600 032.
www.e-consystems.com

Tara - Stereo Vision USB 3.0 Camera

Tara SDK – Linux User Manual

Revision 1.1

12-Jul-2016

Contents

Tara SDK – Linux User Manual

1 Revision History

Rev No	Date	Major Changes	Author
1.1	12-Jul-2016	Initial Draft	Karthikeyan A

2 Introduction

Tara - See3CAM_Stereo is a UVC compliant USB 3.0 SuperSpeed Stereo vision camera from e-con Systems, a leading embedded Product Design Services company which specializes in the advanced camera solutions. Tara is based on MT9V024 stereo sensor from OnSemi and it supports a maximum resolution of WVGA at 60fps over USB 3.0 in uncompressed format. Tara is the latest member of the Stereo Vision family of USB3.0 SuperSpeed camera products launched by e-con Systems.

Tara Software Development Kit(SDK) package, built on OpenCV Image Processing Library is bundled with Tara - Stereo Vision USB 3.0 Camera. Tara SDK includes the commonly used stereo camera functions such as disparity, depth measurement, etc.. and few real time applications implemented in OpenCV. e-con Systems SDK comes with the source code.

Tara is a monochrome camera with the S-mount (also known as M12 board lens) lens holder and pre-calibrated lens pair. The S-mount is one of the most commonly used small form-factor lens mounts for board cameras. Tara has two OnSemi's 1/3 inch MT9V024 image sensors separated by an 'inter-ocular distance' or 'base line' of 60 mm. With USB 3.0 interface to the host PC, Tara can stream WVGA resolution at 60 fps, VGA resolution at 60 fps and QVGA resolution at 60 fps in uncompressed Y16 format. It also has ability to capture still images. Tara is also backward compatible with USB2.0 host ports and does not require any special camera drivers to be installed in the host PC. In USB 2.0 the camera is capable of streaming WVGA resolution at 30 fps, VGA resolution at 30 fps and QVGA resolution at 60 fps in uncompressed Y16 format.

Camera: Tara - Stereo Vision USB 3.0 Camera
Format: Y16
Resolutions Supported:

1 WVGA - (2 x 752) x 480
2 VGA - (2 x 640) x 480
3 QVGA - (2 x 320) x 240

3 Scope

The scope of this document is to highlight the SDK folder structure and applications that are included in the package for Linux Operating System(OS). This document describes the special demo applications that are included in the SDK.

4 Tara SDK Folder Structure

The directory 'Tara_SDK_LINUX_REL_package_xxxx' contains the following

```
Tara_SDK_LINUX_REL_package_xxxx
.
|--------  Documents
|          |-- Tara_Linux_API_Manual.pdf
|          |-- Tara_SDK_IMU_SampleApp_User_Manual.pdf
|          `-- Tara_SDK_Linux_User_Manual.pdf
|
|--------  Prebuilts
|          `-- Ubuntu-14.04
|           |          |-- binary
|           |          |-- binary_x64
|           |          |-- lib
|           |          `-- lib_x64
|           |
|           `-- install.sh
|           `-- uninstall.sh
|
`--------  Source
           |-- common
           |-- examples
           |-- opencv-patch
           `-- README.txt
```

1. Documents

This folder contains user manuals of Tara(See3CAM_Stereo) camera.

a) Tara_Linux_API_Manual.pdf : This document describes Tara namespace used in SDK where all the common functions are put together.

b) Tara_SDK_IMU_SampleApp_User_Manual.pdf : This document describes the usage of an IMU unit(LSM6DS0) integrated with e-con's stereo camera - Tara in application development.

c) Tara_SDK_Linux_User_Manual.pdf - This file (currently reading!!)

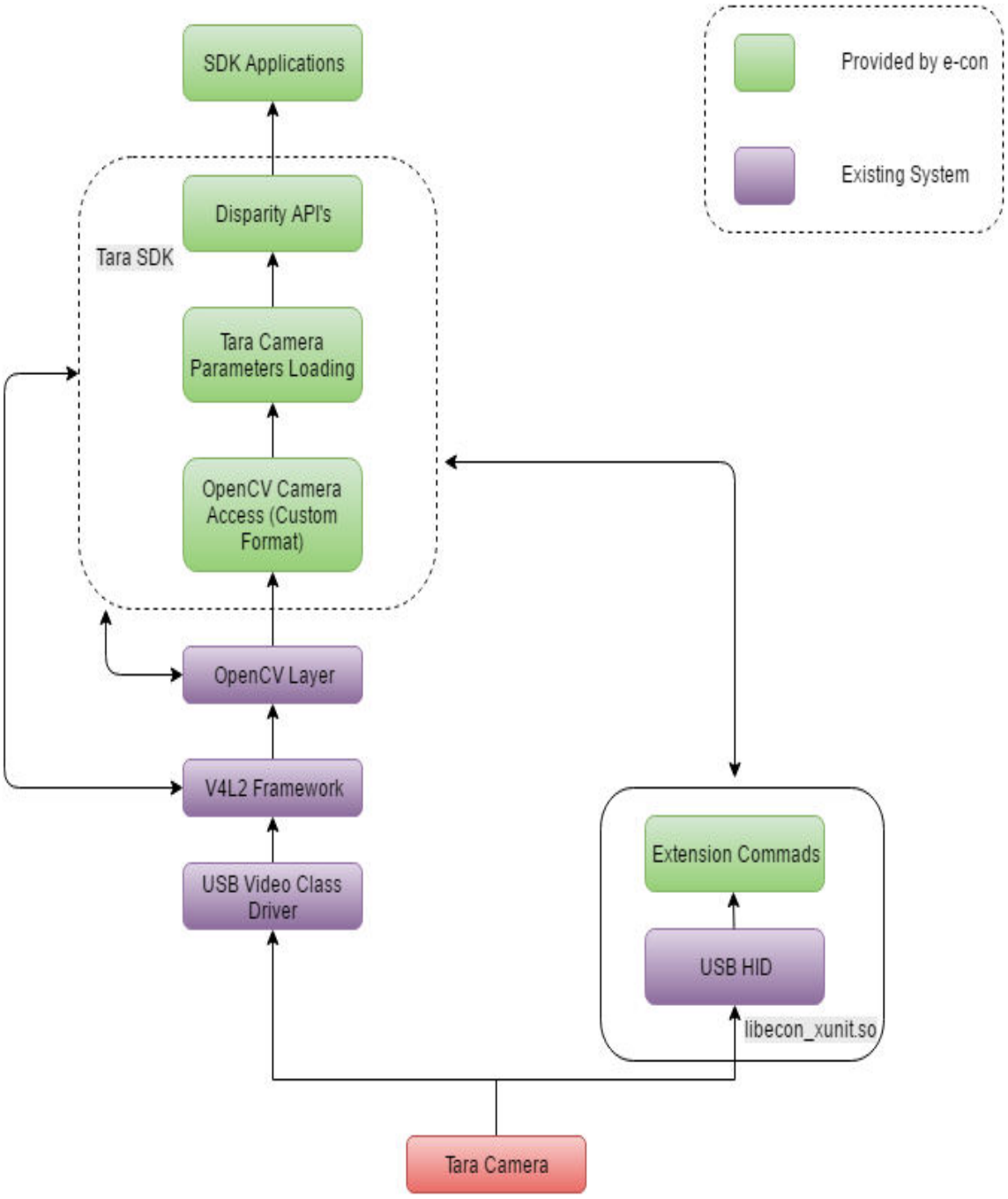
2. Prebuilts

Contains the executable of all the applications in the SDK. The library files of the Tara and Extension unit and the dependency libraries of the OpenCV are also placed. The executable and all the dependencies are build for both 32 and 64 bit respectively in Linux OS Ubuntu-14.04.

3. Source

Contains the Source files for the commonly used functions and sample applications. In the folder named **OpenCV_Patch** the code to be altered in the OpenCV to support custom formats(Y16) is given.

5 Block Diagram



6 Installing Tara SDK Package

In order to install the Tara SDK package, the steps to be followed are

i. Unzip the Tara_SDK_LINUX_REL_package_xxxx.zip using the command. Upon unzipping there will be a tar archive and a README file. Use the README.txt for instructions.

```
$ unzip  Tara_SDK_LINUX_REL_package_xxxx.zip
```

ii. Extract the tar archive using the command. After running the command, there will be a directory named 'Tara_SDK_LINUX_REL_PACKAGE_xxxx'

```
$ tar -xhzvf Tara_SDK_LINUX_REL_package_xxxx.tar.gz
```

iii. Change the location to the Prebuilts folder in the extracted package.

```
$ cd <Extracted Package Location>/Tara_SDK_LINUX_REL_PACKAGE_xxxx/Prebuilts
```

iv. Run the install.sh shell script with superuser privilege. This will install the prebuilt binaries and libraries for the respective LTS version and architecture into the path "/usr/local/tara-sdk" and append the path variables to bashrc and /etc/profile.

```
$ sudo ./install.sh
```

v. Reload the shell to use the modified environment variables by using the command.

```
$ source ~/.bashrc
```

Note : In Tara_SDK_LINUX_REL_package_xxxx : xxxx refers to the release version.

7 Dependencies to be Installed

Install the following dependencies for using OpenCV and to use Point cloud Library.

1. Some general development libraries

```
$ sudo apt-get install build-essential make cmake cmake-qt-gui g++
```

2. libav video input/output development libraries

```
$ sudo apt-get install libavformat-dev libavutil-dev libswscale-dev
```

3. Video4Linux camera development libraries

```
$ sudo apt-get install libv4l-dev
```

4. Eigen3 math development libraries

```
$ sudo apt-get install libeigen3-dev
```

5. OpenGL development libraries (to allow creating graphical windows)

```
$ sudo apt-get install libglew-dev
```

6. GTK development libraries (to allow creating graphical windows)

```
$ sudo apt-get install libgtk2.0-dev
```

7. Udev development libraries (to allow access to device information)

```
$ sudo apt-get install libudev-dev
```

8. Point Cloud Library

```
$ sudo add-apt-repository ppa:v-launchpad-jochen-sprickerhof-de/pcl
$ sudo apt-get update
$ sudo apt-get install libpcl-all
```

This commands will suggest additional dependency packages to be installed. Press 'y' to continue downloading the additional packages.

8 Executing Samples

To run the binaries, Make sure the Tara SDK package is installed (Refer section-6) and the dependencies are installed (Refer section-7) . If those steps are done, then run the applications with the following command,

```
$ sudo -i <BinaryName>
```

For Example, `$ sudo -i TaraCamViewer`
`$ sudo -i FaceDetection`

i. Once the application is launched, list of enumerated camera devices will be displayed. Select the Device ID with the name **See3CAM_Stereo**.

ii. If the stereo Device ID is correctly selected, the resolutions supported by the device will be displayed. Select the Resolution ID to start streaming the camera with that resolution.

iii. The image windows streaming the left and right frames or disparity are shown.
   a) Press '**q/Q/Esc**' on the image window to quit the application.
   b) Press '**e/E**' on the image window to change the exposure of the camera. Exposure ranges from 10 to 1000000 micro seconds. Initially the exposure is set to 15000 micro seconds.

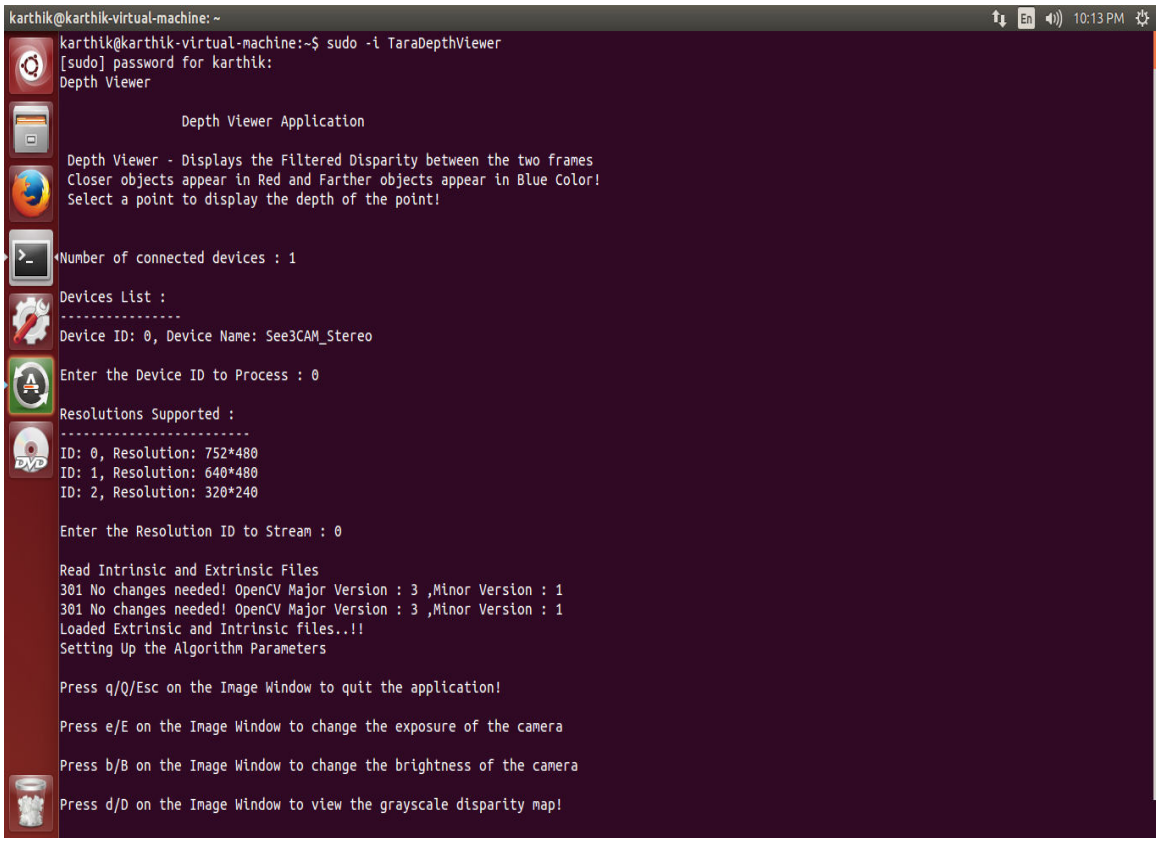
c) Press '**b/B**' on the image window to alter the brightness of the camera. Range is from 1 to 7.

d) Press '**r/R**' on the image window to view the right image of the camera in the samples where the right frame is not displayed.

e) Press '**d/D**' on the image window to view the gray scale disparity map in the Depth viewer and Disparity viewer application.

iv. Refer to the section-12 to know about each of the application included in SDK.

Sample Console:



9 Uninstalling Tara SDK package

In order to uninstall the Tara SDK package, the steps to be followed are

i. Change the location to the Prebuilts folder in the extracted package.

$ cd <Extracted Package Location>/Tara_SDK_LINUX_REL_PACKAGE_xxxx/Prebuilts

ii. Run the uninstall.sh shell script with superuser privilege. This will remove the binaries from the installed location and delete the path variables added to bashrc and /etc/profile.

```
$ sudo ./uninstall.sh
```

10 Developing applications with Tara using OpenCV

If you are going to develop applications on your own using OpenCV, then this step has to be done. To run the prebuilt binaries included in the SDK package ignore this step.

i. Download the latest version of opencv (Version 3.1) for Linux with the following link. (http://opencv.org/downloads.html)

ii. Download the contrib modules additionally from the github repository. (https://github.com/Itseez/opencv_contrib)

```
$ unzip opencv_contrib-master
$ unzip opencv-3.1.0
```

iii. In order to add Y16 format support, Replace the existing cap_v4l.cpp in the folder path: *opencv-3.1.0/modules/videoio/src/* with **e-con's patch file** in the SDK folder path : *Source/opencv-patch/*

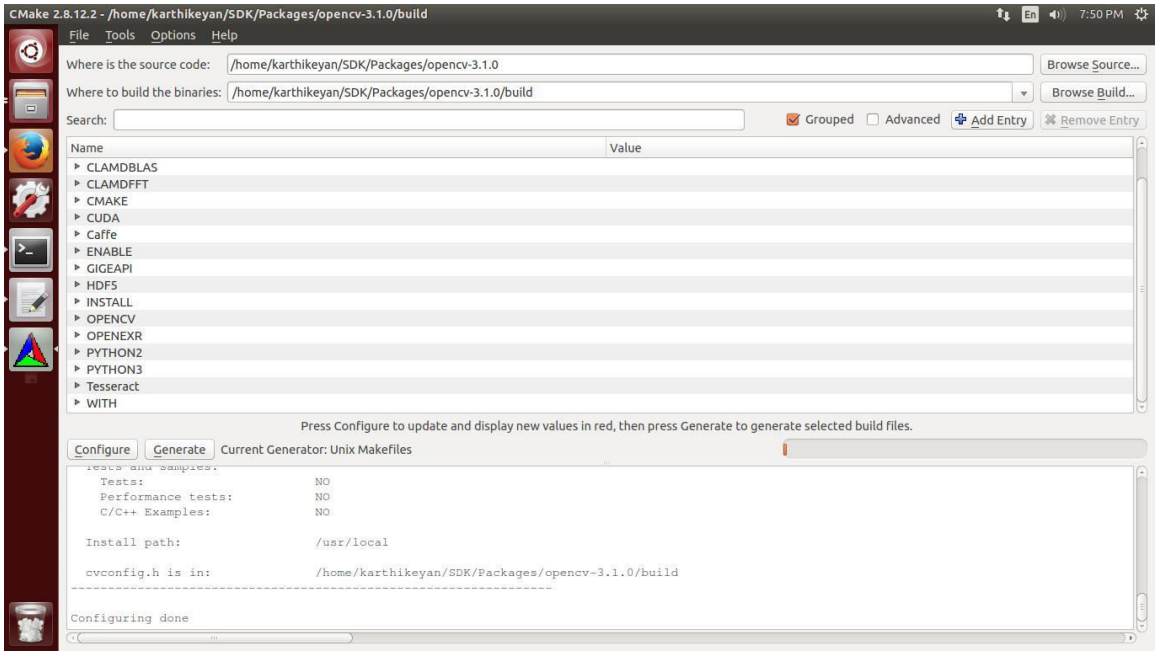
```
$ cd opencv-3.1.0
$ mkdir build && cd build
$ cmake-gui
```

(1) Browse the source code path : <Dir>/opencv-3.1.0
(2) Browse the build path : <Dir>/opencv-3.1.0/build
(3) Press Configure button, with default native compilers set(Unix Makefiles)

Select all the packages you want to use and press again the Configure button. For an easier overview of the build options make sure the Grouped option under the binary directory selection is turned on.

For some of the packages CMake may not find all of the required files or directories. In case of these CMake will throw an error in its output window (located at the bottom of the GUI) and set its field values, to not found constants.

For these you need to manually set the queried directories or files path. After this press again the Configure button to see if the value entered by you was accepted or not. Do this until all entries are good and you cannot see errors in the field/value or the output part of the GUI.



Make the following configurations,

(4) Provide the path for extra modules,

OPENCV_EXTRA_MODULES_PATH : <Dir>/opencv_contrib-master/modules

(5) Uncheck the WITH_LIBV4L option

(6) Check the WITH_TBB & BUILD_TBB option

(7) After configuring, and if there are no errors press the Generate button. Makefiles will be generated, close the GUI and enter the command.

iv. Build and install into the the file system

```
$ sudo make -j4 install
```

This will build and install OpenCV libraries in the location: /usr/local/lib/

11 Commonly used Functions in SDK

(i) The commonly used functions are put into a single class for modularity, in order to reuse them in the examples. Common functions such as loading the calibrated files from the camera flash,

computing disparity map, rectification of frames, camera enumeration and initialization, etc. For further details, refer to the Tara_API_Manual.pdf in the Documents folder.

(ii) Extension unit commands are built as separate library for modularity.

12 Applications in SDK

12.1 Face Detection

Use Case:

Application to measure the distance of the person standing in front of the camera. The application detects multiple faces using the Haar Cascade classifier of OpenCV. If there are any faces in the scene, detected faces are marked with a rectangle and the distance of the detected face from the camera is displayed on the input image.

Note: Application uses the file **"haarcascade_frontalface_alt2.xml"** placed in the folder **"Face"** inside the installed directory to detect the faces.

Environment:

An environment with normal lighting conditions is sufficient. If the scene is too dark try to modify the exposure value.

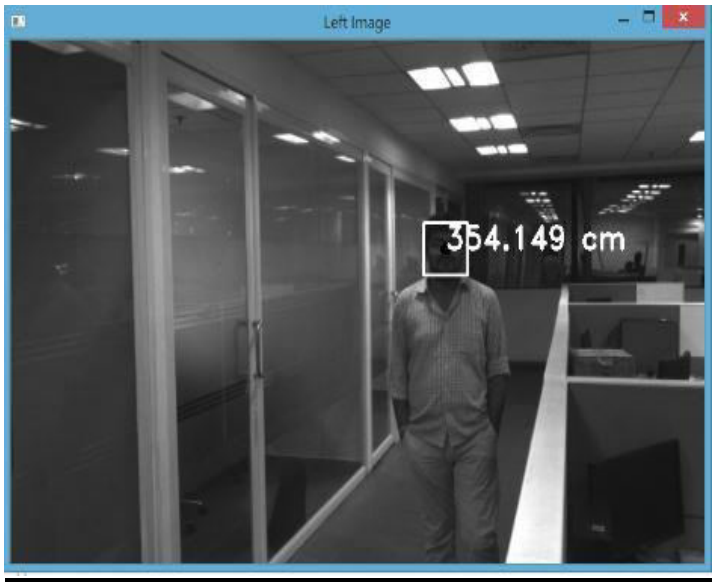
Running the application:

Refer section-8, instead of BinaryName replace it with 'FaceDetection'.

Test Cases:

Full face should be visible in both the frames. Tilt of face to certain extent only will be detected.

Result:



12.2 Height Calibration

Use Case:

In order to measure the height of a person from the camera, there should be some reference value(BaseHeight - Distance of the base from the camera). The Height Calibration application finds that distance. The user has to select a base point, the application calculates the BaseHeight value and save it in a text file “BaseHeight.txt” inside the installed directory’s folder named “Height”. This text file is later used by the Height Estimation Application.

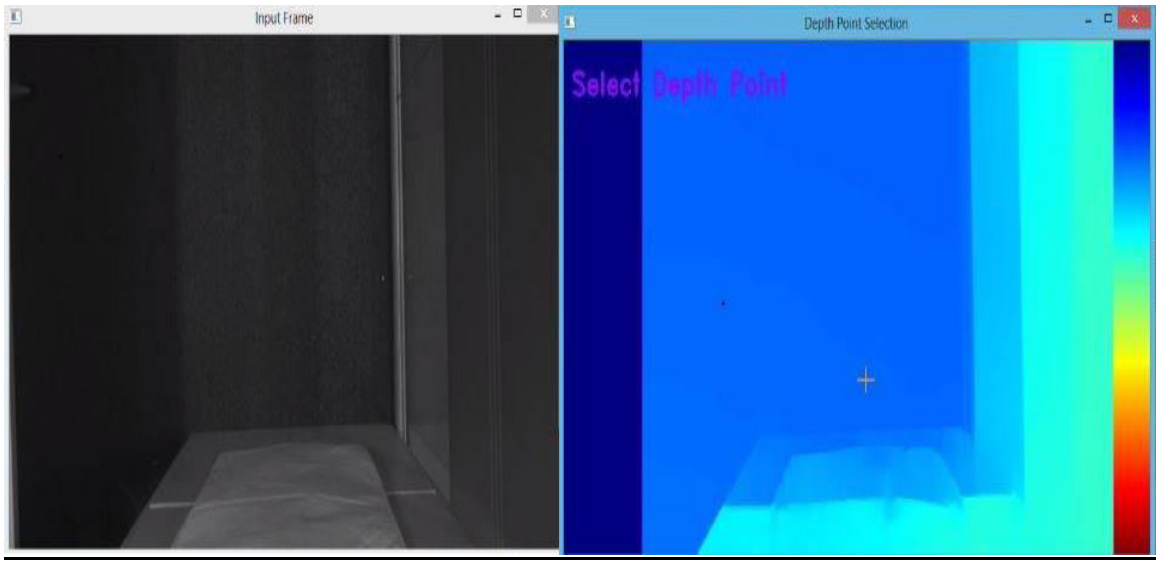
Environment:

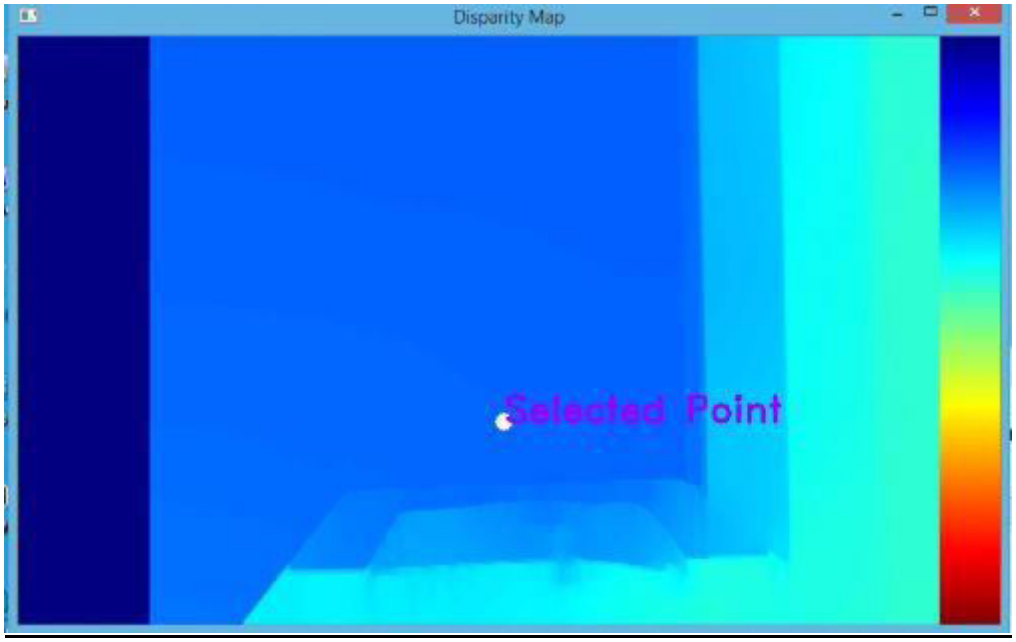
To obtain a perfect depth measurement of the base, use some textured element placed in the base and select those points to measure the depth. Depth deviation directly depends on the texture of the point selected.

Running the application:

Refer section-8, instead of BinaryName replace it with ‘HeightCalibration’.

Result:





12.3 Height Estimation

Use Case:

Estimates the height of a person standing under the camera, the head height is estimated from the disparity map and the human height is given with the reference taken from the base using the height calibration application.

Note: Application uses the file **"BaseHeight.txt"** placed in the folder **"Height"** inside the installed directory to estimate the height.

Environment:

The depth is estimated from the disparity map of the person standing under the camera. The point which is used as the head height is selected by scanning for the lowest depth in the 1/3rd of the image. The head of the person should be in the 1/3rd of the image, this is done to avoid scanning of full image. Its always the better way to have the camera 0.5m above the head, to have a clear disparity map.

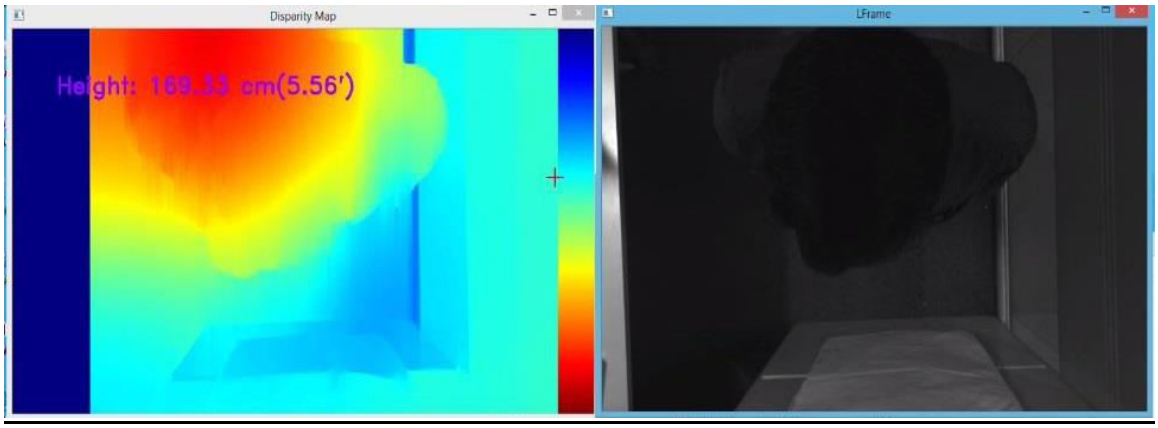
Running the application:

Refer section-8, instead of BinaryName replace it with 'HeightEstimation'.

Test Cases:

Validation such that the scene has a person in it is not done. The head height is subtracted from the base height to know the height of the person. Hence BaseHeight file under the Height folder is mandatory, otherwise the application will through an error. The height of the person depends on the base height measured.

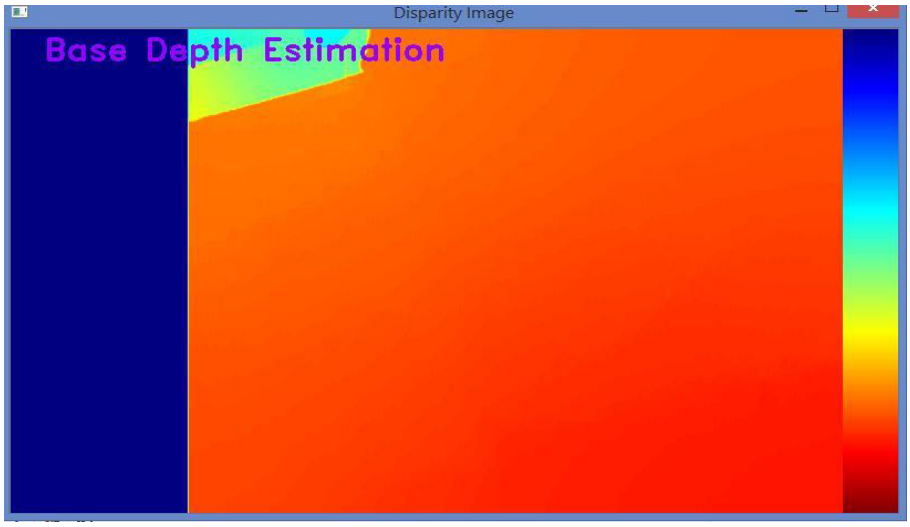
Result:



12.4 Volume Estimation

Use Case:

Estimates the Volume of the box placed in the scene using the simple Edge detection technique. Initially the distance of the base is estimated by averaging the depth taken at a point in few frames. To have an accurate depth measurement, the base should have a proper texture.



With the reference to the base height estimated in the previous stage the height is found.

Environment:

To have an accurate depth measurement, use some textured element placed in the base for the base depth estimation stage. Depth deviation directly depends on the texture of scene. This is important since the height of the box is estimated only using the base height measured.

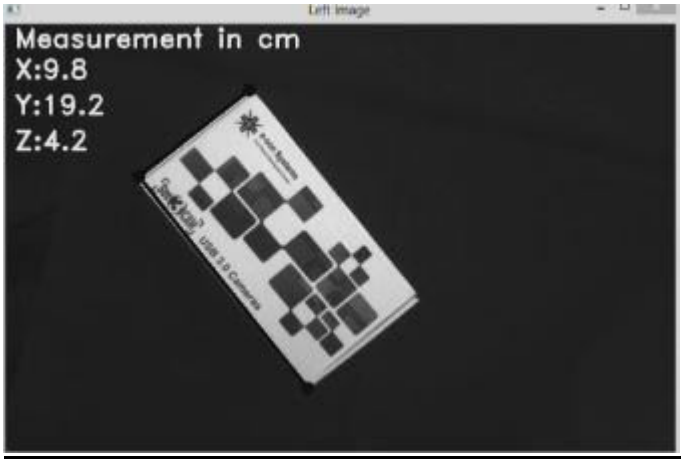
Running the application:

Refer section-8, instead of BinaryName replace it with 'VolumeEstimation'.

Test Cases:

The scene should have only the box to be measured, since the application does not validate for a rectangular object.

Result:



12.5 TaraCamViewer

Use Case:

OpenCV application to stream the Tara camera. This Stereo vision camera provides two synchronized sensor frame data interleaved side by side to the host machine over USB 3.0 interface. The TaraCamViewer application uses the Tara layer(libecon_tara.so) to grab the interleaved frame using OpenCV and then splits the interleaved data into Left and Right camera output separately.

Environment:

Under any environment conditions, the application streams the scene whatever it is viewing.

Running the application:

Refer section-8, instead of BinaryName replace it with 'TaraCamViewer'.

Result:

12.6 TaraDepthViewer

Use Case:

Application to measure the depth of an user selected point. Disparity map refers to the apparent pixel difference or motion between the left and the right image. The application displays the disparity map, left and right frame. When a point is selected in the disparity map, the depth of the point from the camera will be displayed.

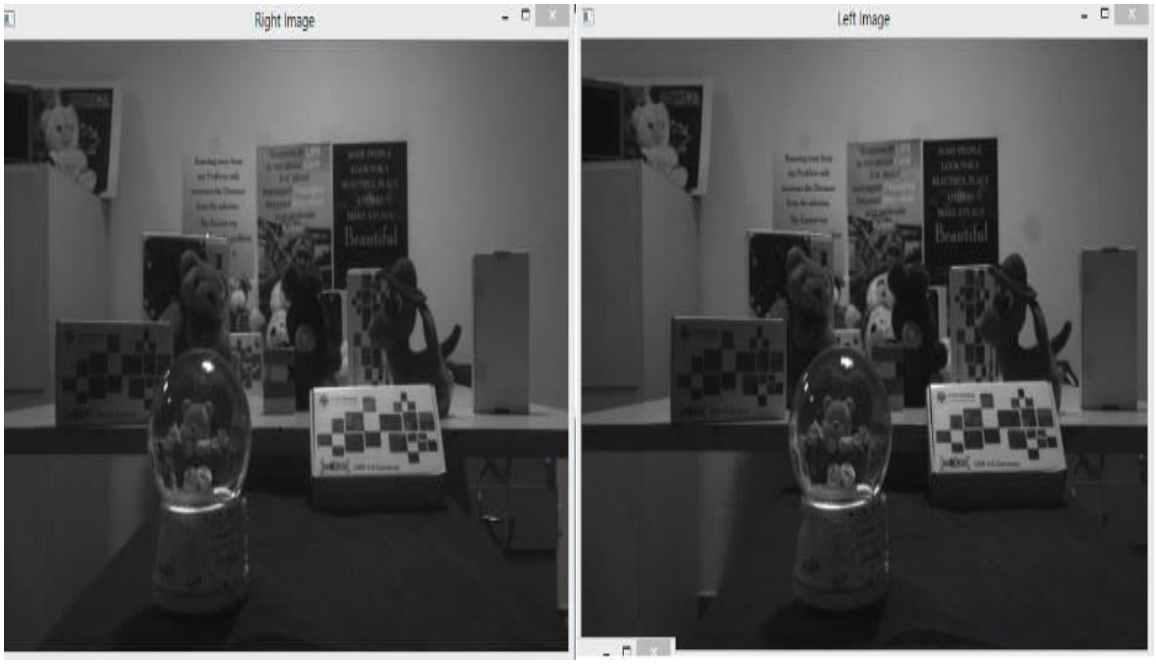
Environment:

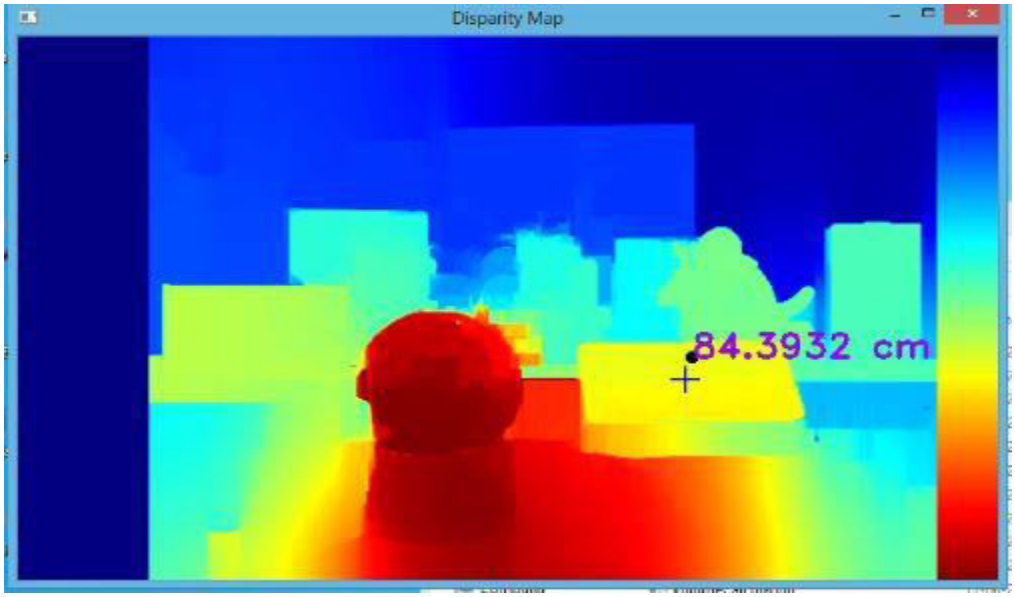
To have an accurate depth measurement, the point selected should be textured. Depth deviation directly depends on the texture of the point selected.

Running the application:

Refer section-8, instead of BinaryName replace it with 'TaraDepthViewer'.

Result:





12.7 TaraDisparityViewer

Use Case:

Application to display the disparity map measured from the pair of stereo images. Disparity map refers to the apparent pixel difference or motion between the left and the right image. The application displays the Left, Right and Disparity map without any filtering being applied.

Environment:

To have a better disparity map, the scene should have textured objects.

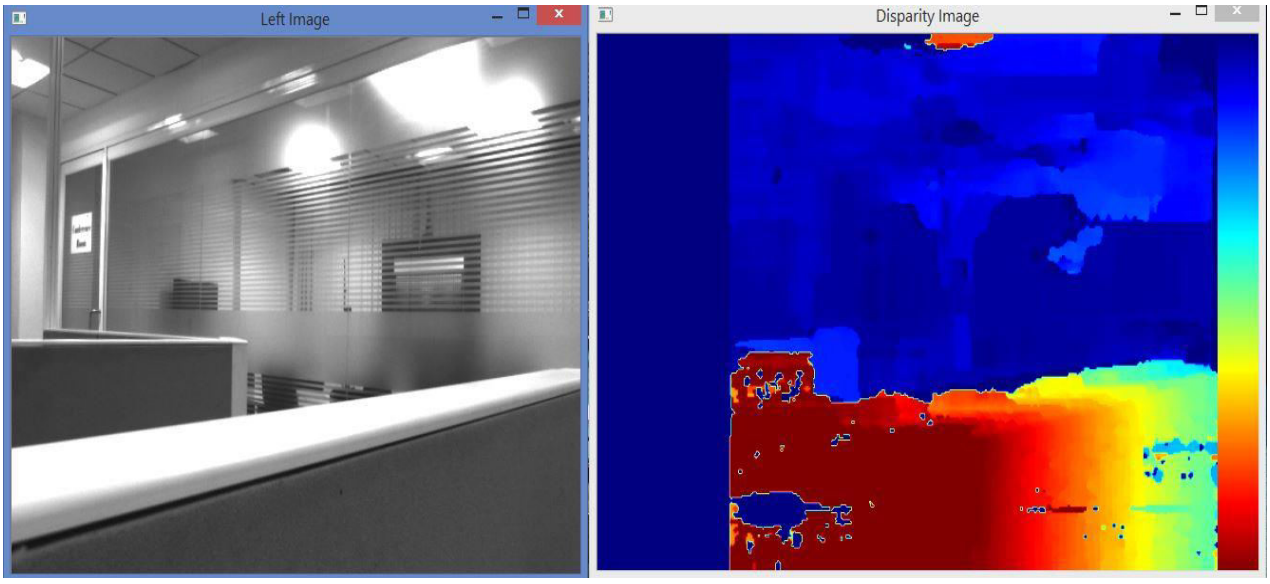
Running the application:

Refer section-8, instead of BinaryName replace it with 'TaraDisparityViewer'.

Test Cases:

The objects closer to the camera appear red in color whereas the farther objects appear in blue color. The scale of color map in the right side of the disparity map is relative to the distance of the object from the camera(varies from red to blue).

Result:



12.8 Point Cloud

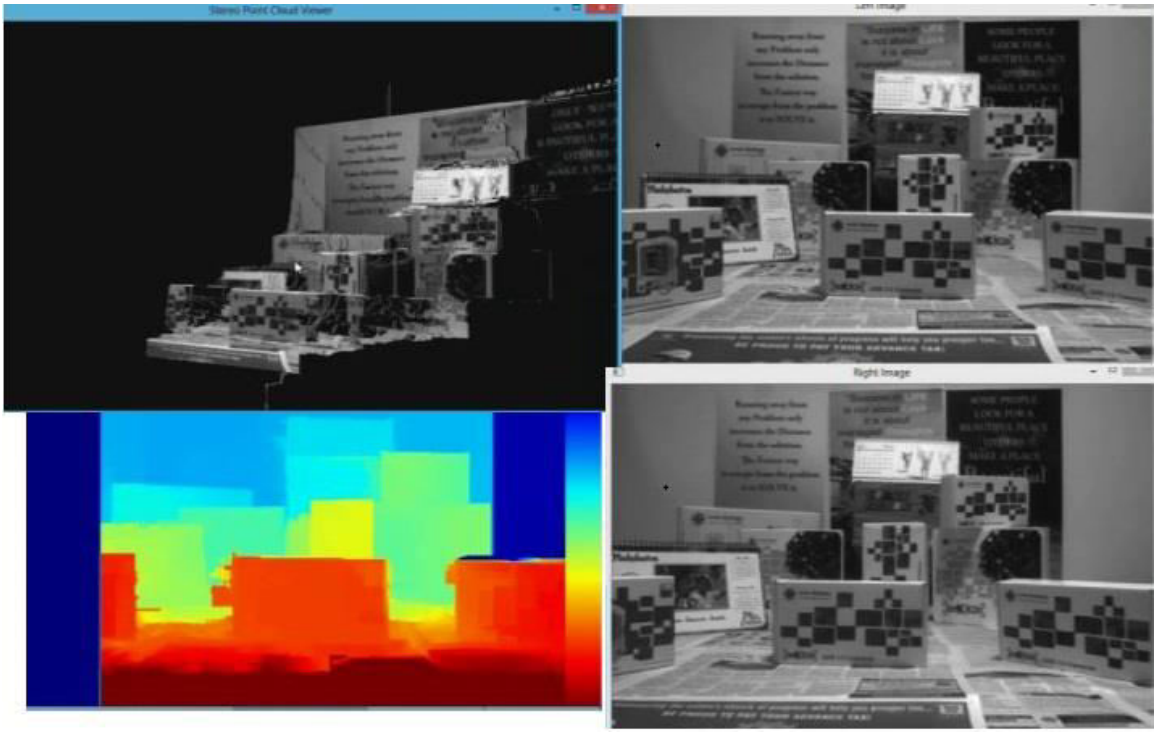
Use Case:

Application to display the Point Cloud sample of the scene. Using the disparity map computed from the stereo pair images, a 3D point cloud is rendered in the window.

Running the application:

Refer section-8, instead of BinaryName replace it with 'PointCloud'.

Result:



12.9 IMU Application

Use Case:

Application to illustrate the IMU unit LSM6DS0 that is integrated with the stereo camera-Tara. The IMU sample application is a basic example demonstrating the rotations of camera around x-axis and y-axis. The output rotation angles calculated from the IMU values are limited to the range from -90 to +90 degrees for illustration. It displays a window where the rotation of the device is plotted for all the three axis respectively.

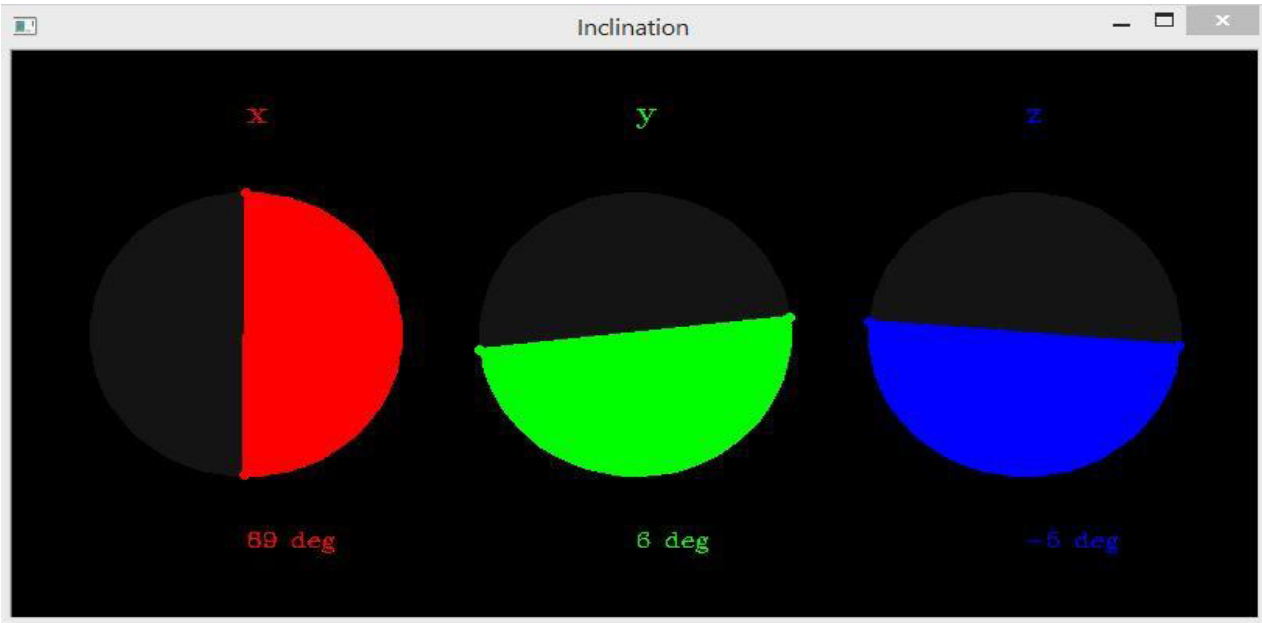
Environment:

This IMU application doesn't depend on the scene, its only based on the camera rotation around x and y axis, so there is no need for any specific environment.

Running the application:

Refer section-8, instead of BinaryName replace it with 'ImuApplication'.

Result:



13 Known Issues

The Tara Stereo camera also has an option to output the 10 bit monochrome format. The format is given as RGB-24 from the camera.
Pixel arrangement : Byte 1 - X X X X M1 M0 S1 S0 Byte 2 - M9 M8 M7 M6 M5 M4 M3 M2 Byte 3 - S9 S8 S7 S6 S5 S4 S3 S2
Key - X - Don't care , M - Left , S - Right.

The UVC supports BGR3 format only from the kernel version 4.0. Therefore In order to get the 10 bit data, the kernel has to be updated to the higher versions above 4.0.

Commands to update the kernel:

1) Update the kernel version to 4.2 to support the RGB format in uvc layer

$ sudo apt-get update

$ sudo apt-get install linux-image-generic-lts-wily

2) Reboot the system to reflect the changes.

```
$ sudo reboot
```

14 Conclusion

This document provides the details of Tara SDK folder structure and applications provided in the Linux release package.