Qualcomm Technologies, Inc.

Dual Camera Assembly and Calibration Guide

80-N5126-3

June 1, 2015

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

Revision history

Revision	Date	Description
A	Jun 2015	Initial release

Contents

Figures

Tables

1 Introduction

1.1 Purpose

This document guides module vendors with procedure to manufacture and calibrate dual camera modules for use with Qualcomm Technologies Inc. (QTI) software supporting enhanced functionality with such hardware.

1.2 Conventions

Function declarations, function names, type declarations, attributes, and code samples appear in a different font, for example, `#include`.

Code variables appear in angle brackets, for example, `<number>`.

Commands to be entered appear in a different font, for example, **`copy a:*.* b:`**.

Button and key names appear in bold font, for example, click **Save** or press **Enter**.

1.3 Technical assistance

For assistance or clarification on information in this document, submit a case to Qualcomm Technologies, Inc. (QTI) at https://support.cdmatech.com/.

If you do not have access to the CDMATech Support website, register for access or send email to support.cdmatech@qti.qualcomm.com.

2 Dual Camera System Requirements

This chapter covers the requirements for dual camera assemblies. The purpose is to provide a top-level view of the key mechanical, optical, and electrical characteristics to align efficiently with the QTI's dual camera architecture. Aspects such as the assembly frame, shielding, and connector/flex are left to the module integrator and OEMs. They are not relevant to the compatibility of the assembly to interface to the QTI architecture, if dimensional stability is maintained from manufacture through final target device assembly.

Below two types of dual camera assemblies are described. First are asymmetric dual camera assemblies that use a higher resolution primary sensor, and a lower resolution auxiliary sensor that is used to generate the depth map. Second are the symmetric camera assemblies that use equivalent sensors and optics, either in a Bayer+Bayer, or Bayer+Mono color filter configurations.

2.1 Asymmetric dual camera assembly camera requirements

Figure 2-1 shows the top-level view of the asymmetric dual camera assembly with a 1/3" primary camera. The module is composed of two cameras and a Nonvolatile Memory (NVM) to store calibration coefficients. The primary camera is Auto Focus (AF) with a bayer Color Filter Array (CFA).

A camera separation of 20 mm is recommended for good depth mapping performance. If the primary use case is instantAF (IAF), the separation can be as little as 10 mm to allow for a more compact assembly.

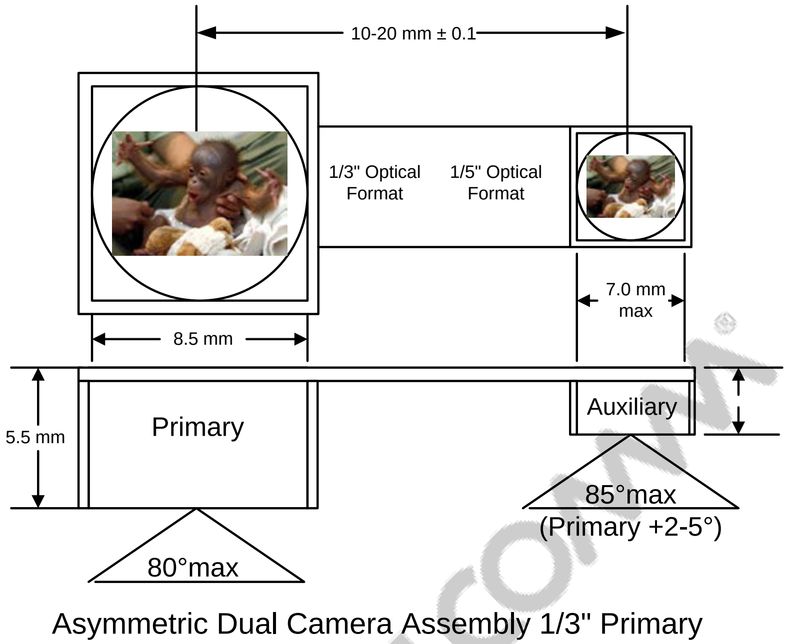


Figure 2-1 Asymmetric dual camera module mechanical guidelines

The optical requirements are such that the Field of View (FOV) for the auxiliary camera to consider such factors as:

- Manufacturing tolerances of both cameras
- Change in FOV for primary camera over focus range
- Changes in FOV for both cameras over manufacturing and environment
- Overlap at minimum distance
- Maximum resolution loss intended at minimum working distance

The recommendation of 80° maximum for primary camera is based on typical optical designs to meet required module heights while not incurring excessive distortion, Logic Cell Array (LCA), shading, or other aberrations. More aggressive designs will require engagement and study with QTI.

Resolution requirements for the auxiliary camera to support DDM generation are proportional to primary camera resolution. Our analysis and empirical results indicate that the resolution of the auxiliary sensor can reduce up to ¼ (in both dimensions) of the main sensor without visible reduction of quality. For the 13 MP dual camera, Figure 2-1, the auxiliary sensor is operated in a 2x2 binning mode for an output of 800x600 pixels at 30 fps. For the 21 MP dual camera module, the auxiliary sensor is operated at full resolution of 1600x1200 pixels at 24 fps.

Table 2-1 Asymmetric dual camera module sensor selections

Configuration	Primary	Auxiliary
13 MP+2 MP	OV13850	OV2685
21 MP+2 MP	IMX230	OV2685

2.2 Symmetric dual camera assembly camera requirements

Figure 2-2 shows the top-level view of the symmetric dual camera assembly with 1/3” sensors. The module is composed of two cameras and an NVM to store calibration coefficients. The primary camera is AF with a bayer CFA. The secondary camera is AF and may have either a mono or bayer sensor.

Camera separation recommendations are based on intended primary use case. A camera separation of 20 mm is recommended for good depth mapping performance. If the primary use case is IAF, the separation can be as little as 10 mm. The Fusion use case is best with minimum separation between the modules. This accommodate acceptable IAF performance.

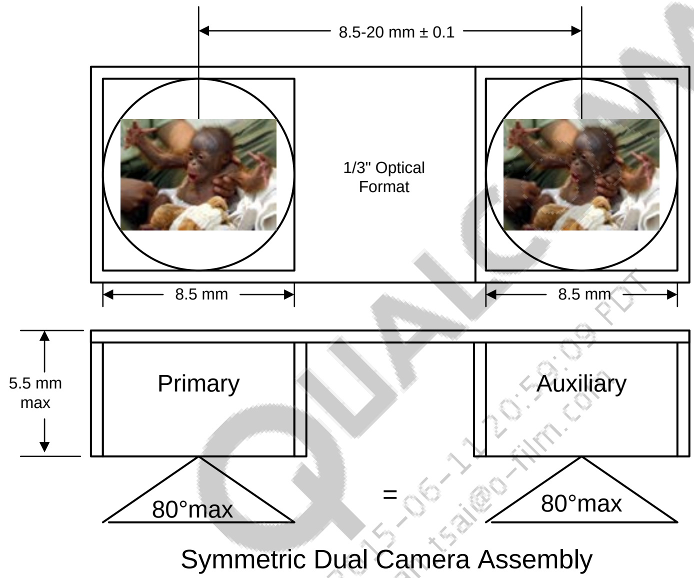


Figure 2-2 Symmetric dual camera module mechanical guidelines

The recommendation of 80° maximum for primary camera is based on typical optical designs to meet required module heights while not incurring excessive distortion, LCA, shading, or other aberrations. More aggressive designs can be considered, but requires engagement and study with QTI.

Table 2-2 Symmetric dual camera module sensor selections

Configuration	Primary	Auxiliary
13 MP+13 MP	OV13850	OV13850/OV13851
8 MP+8 MP	OV8865	OV8865

2.3 Camera alignment requirements

The alignment requirements for placing the two cameras on one board. It is important to take measures when designing a dual camera assembly to stay within the camera alignment requirements presented in this section. Two sets of requirements are presented:

- Static tolerances – These tolerances are before calibration and determined by the vendor capability.
- Dynamic tolerances – These tolerances are changes in the camera alignment after the module is calibrated.

2.3.1 Static tolerance requirements

The definition of the static tolerance is the alignment of the modules before calibration. The static tolerance requirements are made up of three translational and three rotational tolerances.

The module is designed to meet the three translational and three rotational tolerances covered in this section. Section 2.3.2 covers how these tolerance limitations tested during manufacturing tests.

Figure 2-3 shows the ideal arrangement of the two cameras, where the sensor planes are ideally aligned with the camera lens optical axis. The two cameras are on the same plane, separated by distance B, and have no vertical shift along the X-axis. They do not have any rotational differences as defined by yaw, pitch, and roll relative to each other in the ideal alignment.

The ideal arrangement is not feasible and so we must allow for rotation and shift.

Rotational tolerances for pitch, yaw, and tilt are 1.0° maximum. The translational tolerance for shift is ±0.3 mm. The rotational and translational tolerances are held tighter if process capability allows.

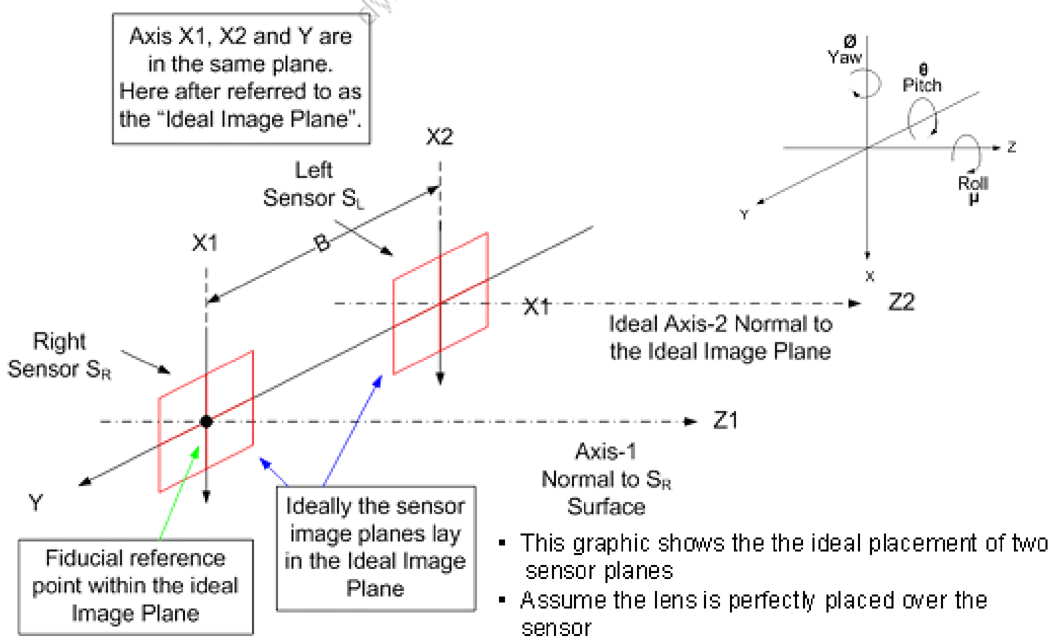


Figure 2-3 Dual camera module alignment tolerances

2.3.2 Dynamic tolerances requirements

All translational and rotational changes after 3D static calibration held to tight tolerances. During manufacturing, the qcaldc calibration tool is run for each module. This program generates the correction matrix coefficients used to remove the static rotational differences between the two cameras.

After the calibration process is complete, it is possible that the camera altered in such a way to cause the camera alignment to change. These types of changes can be caused by:

- Module bent after being dropped
- Module warped after handset assembly handset

Measures must be taken to prevent these module changes from happening.

Assuming the types of changes are under control and held to near zero, the next most important factor to consider is the lens dynamic tilt in the yaw and pitch directions. For AF lenses, the optical axis of the lens barrel can change randomly. The causes of such random changes in the optical axis include:

- Lens barrel moved during the AF operation
- Vibration
- Orientation of the camera, e.g., pointing up, down, horizontally, or any direction in between

The requirements for dynamic tilt are:

- Maximum allowable dynamic variations for both yaw and pitch of each separate camera is ±0.16° about its optical axis
- Maximum allowable pitch and yaw differences between the two cameras is ±0.50° when both lenses are focused at the same distance between 50 cm and infinity

The translational tolerance for shift is ±0.3 mm.

For any specific scenario of tolerances, the actual performance depends the design factors. The Dual_Camera_Calibration_Parameters.xlsx. file at the Attachments portion of this document. Fill out the spreadsheet and return it to QTI via your TAM for further guidance.

2.4 Dual camera module electrical considerations

The module is composed of two cameras and an NVM to store calibration coefficients.

Each camera module has an independent I2C interface with the MSM. This allows identical AF controllers and sensors to be used in the two modules.

One camera module is designated the master, and one camera module is designated the slave. The master camera module outputs a vertical frame synchronization signal. The slave camera module has an input pin, which is the vertical frame synchronization input. These two lines typically be tied together either internal to the dual camera assembly, or connected at the board-level.

- Each cameras output interfaces to the MSM over MIPI CSI-2 lanes
- NVM is attached to the I2C line of either camera

3 Dual Camera Calibration

This chapter describes the procedure for mechanical placement and orientation of the dual camera calibration components for use with the supplied dual camera calibration and depth map software libraries. The methodology described here is suitable for automated assembly line calibration of dual camera modules. A validation procedure described is based on computing metrics and assigning pass/fail thresholds.

Two separate charts are employed for dual camera calibration, a nonplanar L-chart that is composed of two perpendicular checkerboards, and a planar checkerboard chart. The L-chart is used to compute the projective calibration component, and the planar chart is used for geometric correction. The calibration charts used to create the test scenes, chart placements relative to the dual module containing the left and right cameras, and a brief overview of the subsequent calibration processes are described. The given mechanical procedure for calibration serves as a guideline for capturing images to be used with QTI's dual camera calibration library.

3.1 Calibration procedure

The procedure for calibrating a dual camera system using QTI's calibration library consists of obtaining a suitable nonplanar calibration chart. Orienting the camera module such that the FOV coincides with the chart, and capturing and saving the left and right chart images for input into the calibration software. We refer to the two cameras equivalently as left and right, reference and auxiliary, or main camera and auxiliary interchangeably with the understanding that for asymmetric dual systems the reference and/or main camera contains the higher resolution sensor.

Obtaining a dual image pair with the test module of the planar chart is the second step in the calibration process. Images captured subsequent to capture of the L-chart with the exact same focal length settings. The planar geo chart placed next to the L-chart as an additional precomposed factory calibration scene. The module can either be translated to this scene along a conveyor belt or rotated such that the new chart is viewed. The setup is shown in Figure 3-1 where the arrow implies that the module is moved to view scene two.

During the calibration image capture process the same focus settings must be used on both the L-chart and flat chart.

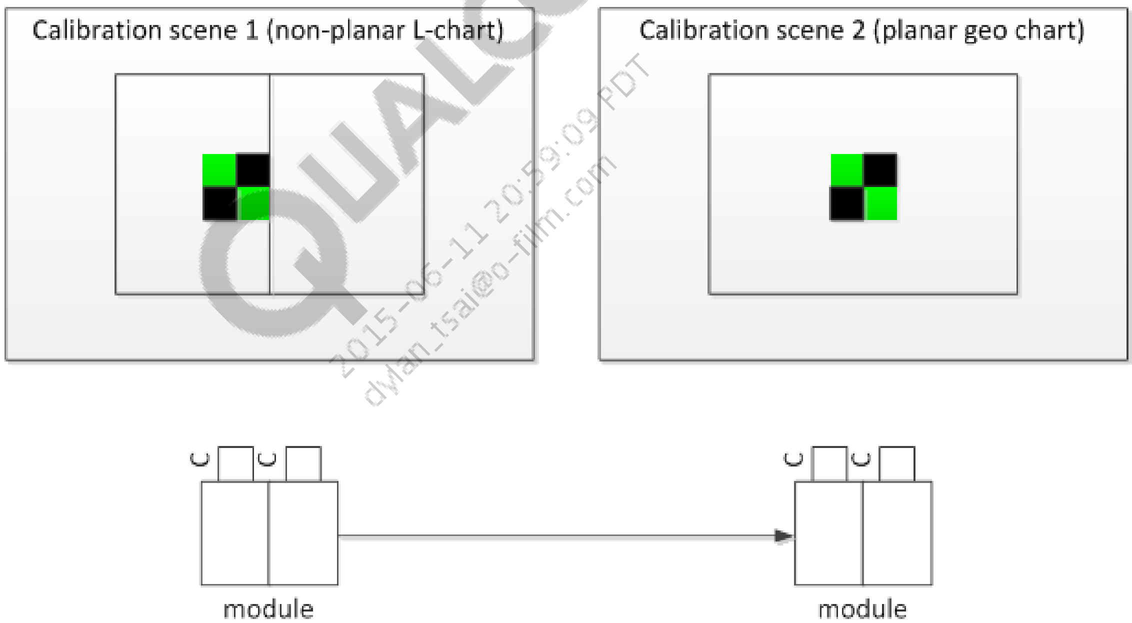


Figure 3-1 Calibration scene diagram

3.2 Calibration scene 1 (nonplanar L-chart)

3.2.1 Construct the test chart

The test chart provides the calibration software with an orthogonal reference or measurement coordinate frame. This frame of reference is used to identify and associate position correspondences between the left and right images.

1. Print the calibration left and right checkerboard chart images on a quality printer. Attach them to firm surfaces, such as straight cardboard or other rigid substrate.

   Affix the two charts to the corner of a room (two-walled intersection) or equivalent perpendicular surfaces. It is also possible to construct a mobile (nonplanar) calibration rig as shown in the following picture.

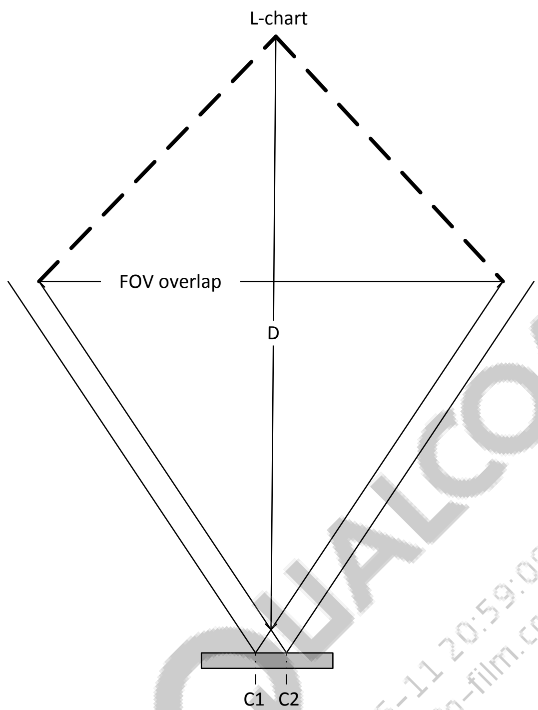
   The test chart consists of two planes sized 2 x 3 feet containing 12 x 18 black and white checkerboard squares 2 inches in length.

2. Once the chart is constructed, measure the size of the squares with a ruler and record the length for input to the calibration software.
3. Place the two planes at right angles so that the joining line is devoid of any interstitial gaps.

   The left plane contains two green patches that are used to identify the coordinate system origin. It is important that the charts are set up at right angles to each other to furnish the calibration software with a sufficiently orthogonal reference frame. The charts must be oriented properly in the vertical direction as they are not symmetric. The left and right charts are not interchangeable and must be placed properly.

3.2.2 Set up the test scene

1. Place the dual camera pair at a distance so that the chart exclusively occupies the FOV of both the auxiliary and reference cameras as illustrated.



   The green patches, which serve as the origin for the world coordinate system, and positioned in the center of the camera FOVs.

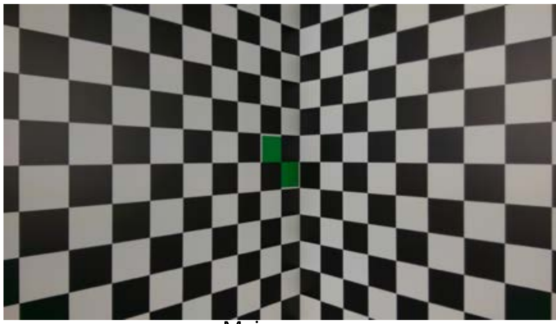
   The place the module equidistant from the two perpendicular planes, ideally around 65-85 cm from the green square.

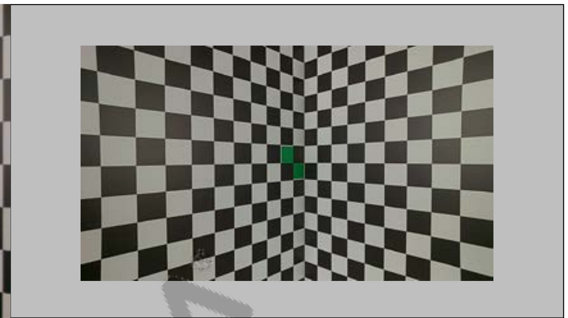
2. Set position tolerances to ±2.5 inches in X, Y, and Z.

   Tolerances are sufficient to accommodate a high degree of flexibility when performing calibration as part of assembly line automation.

3. Set the baseline rotation < 5°.
4. Align the two green squares roughly in the middle of the image.
5. Ensure that the chart is illuminated at +500 lux (or the mean value of the intensities at the 8-bit raw image stage around the center about 50 lux). Use D65 illuminant.

This image is an example of a dual image pair that satisfies this criteria.


Main camera


Auxiliary camera

3.2.3 Perform calibration

Once the scene setup is complete, refer to the qcaldc User Guide, Chapter 4 for the calibration tool instructions and the tool itself.

1. Set up the chart as described in Section 3.2 and Section 3.2.2 and record the chart size parameters.
2. Focus the reference/main camera on the green square and record the focal setting. Capture the L-chart images.
3. Maintain the same focus setting for the flat chart as used on the L-chart. It is important that no focus change is allowed. Identical settings have the added advantage of eliminating dynamic lens errors and require only single focus adjustment during the calibration process.
4. Store the calibration parameters for input to the calibration software.
5. Run the calibration software to complete dual camera calibration.

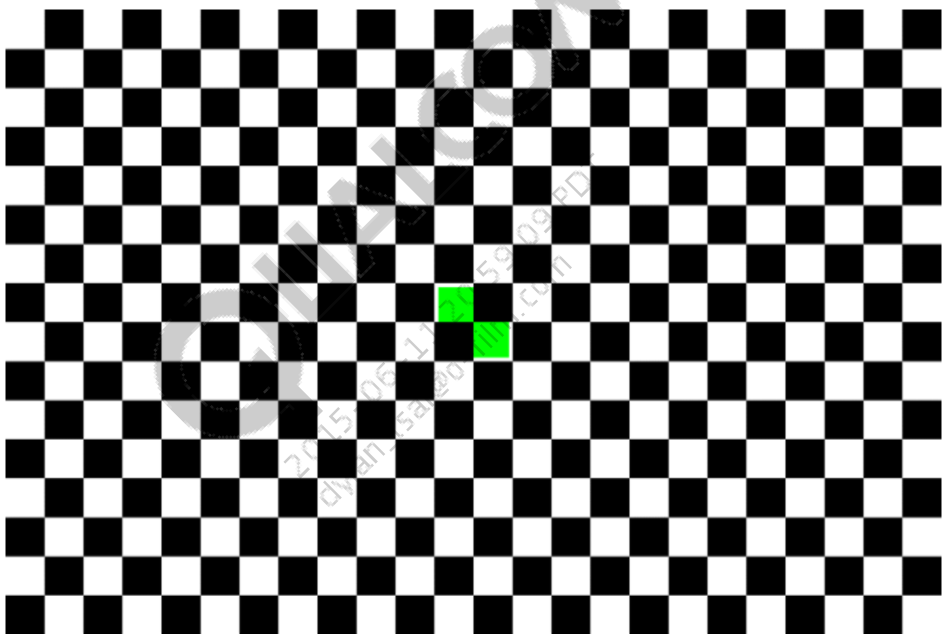
3.3 Calibration scene 2 (planar geo chart)

3.3.1 Construct the test chart

The test chart provides the calibration software with correspondence for lying on a known plane. This frame of reference is used to identify and associate position correspondences between the left and right images.

1. Print the calibration left and right checkerboard chart images on a quality printer.
2. Attach the images to firm surfaces, such as straight cardboard, foam board, or other rigid substrate.

   The test chart consists of 16 x 24 checkerboard squares of size 1.5 inches or 38.1 mm. The center white squares are marked in green.



3.3.2 Set up the test scene

1. Set the camera rig 15 to 30 inches from the chart ensuring that only the chart is present in the left and right camera FOVs.

   Ensure that both cameras only see the chart in their FOV (it is acceptable if they do not see the full chart). The green patches, which serve as the origin for the world coordinate system, positioned in the center of the camera FOVs.

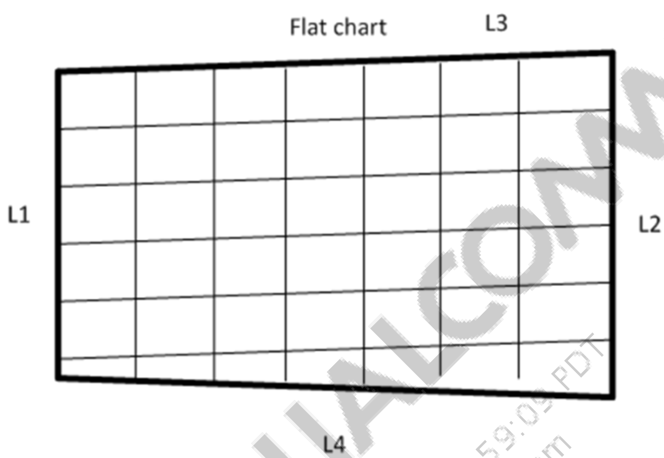
   The module placed equidistant from the two perpendicular planes, ideally 2 feet from each plane (horizontally) and 1.5 feet from the bottom of each plane (vertically).

2. Set position tolerances to ±2.5 inches in X, Y, and Z.

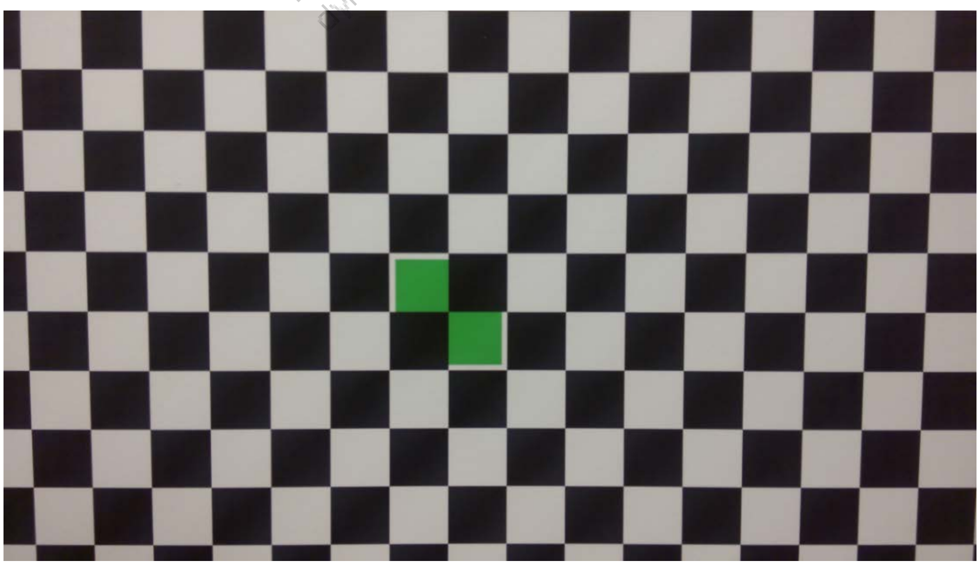
   Tolerances are sufficient to accommodate a high degree of flexibility when performing calibration as part of assembly line automation.

3. Set the baseline rotation < 5°.
4. Align the two green squares roughly in the middle of the image.

   For the flat chart scene, ensure measure the length of vertical lines in pixels on each side of the chart and ensure that the difference is less than 5%. This will ensure that a yaw angle is less than 3°. The value of 1-L1/L2 < 5% and 1-L3/L4 < 5%. This ensures that yaw induced tilt is small enough for calibration to operate.



5. Maintain the same focal position used in the previous capture of the L-chart.
6. Ensure that the chart is illuminated at +500 lux (or the mean value of the intensities at the 8-bit 8-bit raw image stage around the center and about 50 lux). Use D65 illuminant.



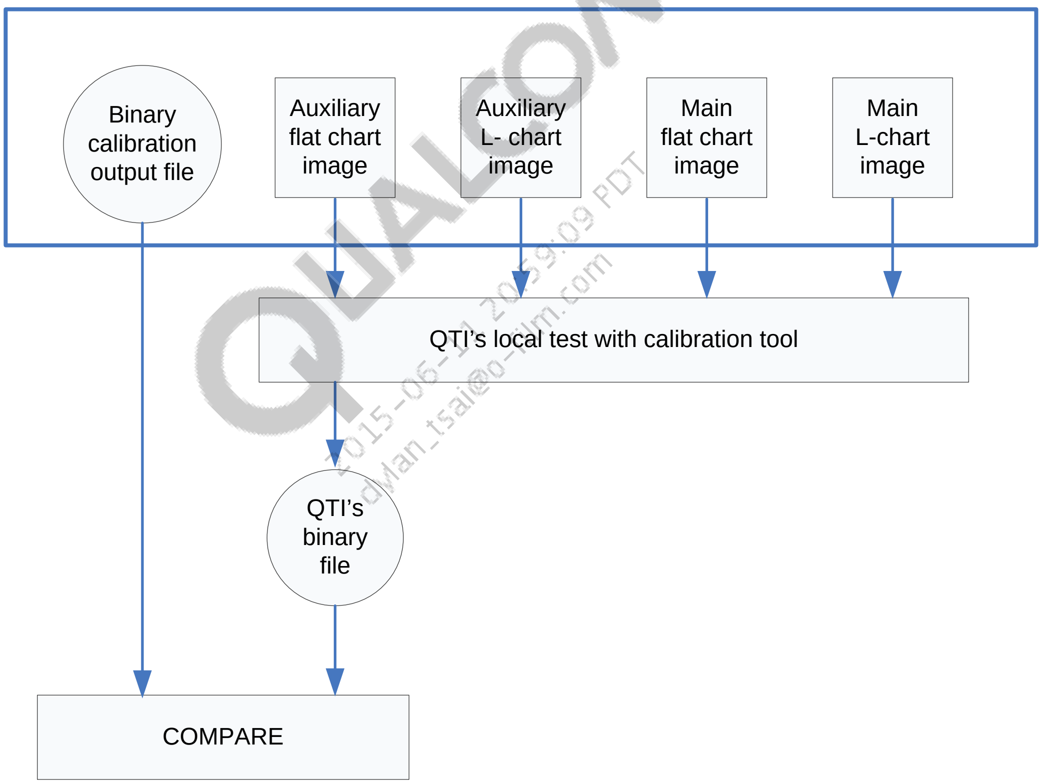
3.3.3 Perform calibration

Once the scene setup is complete, refer to qcaldc User Guide, Chapter 4 for the calibration tool instructions and the tool itself.

1. Print and install the geometric chart as described in Section 3.2.1 and Section 3.2.2.
2. Capture the reference/auxiliary images (color) of the geometric chart (note that the camera must be at the same focal position as the dual camera calibration).

3.4 Validation of calibration procedure

Validation description and procedure provided in Figure 3-1. Calibration images are input into the calibration tool locally at QTI and output binary generated to compare against the customer generated binary.



Validation of calibration procedure is owned by QTI. Module vendor should provide following information to QTI when either calibrating modules for a new project or after making any changes to their existing QTI validated calibration procedure setup.

- Binary calibration file and execution log from the qcaldc tool as described in Section 4.1.1.
- Auxiliary flat chart image
- Auxiliary L-chart image
- Main flat chart image
- Main L-chart image

4 qcaldc User Guide

The qcaldc tool is a command-line Windows program used to calibrate a dual camera sensor module and generate binary data used in processing to correct the module. Because it is difficult and expensive to mount two camera sensors directly parallel to each other, correction is needed. BMP images of the calibration chart from each sensor are fed into qcaldc, which calculates the correction data. These images, along with command-line arguments and a configuration file, are then inputs and a binary file is the output. The qcaldc tool also checks the dual camera sensor module for vertical misalignment, scale, and predicted distance accuracy allowances.

4.1 Installation

4.1.1 Executable

The qcaldc tool uses C code to perform calibration. The C code is packaged in the form of a Windows dynamic library (qcaldc.dll). There is an executable front-end (qcaldc.exe) that accepts the name of the configuration file (qcaldc.cfg) and then invokes the backend calibration software library (qcaldc.dll).

4.1.2 Prerequisites

The qcaldc tool uses Microsoft Visual Studio 2010 and requires the latest version of Microsoft Visual C++ 2010 Redistributable Package (x86). To check if this package is installed on the PC, follow these instructions:

1. Go to Start→Control Panel→Programs→Programs and Features.
2. Search for C++ in the Search Programs and Features window on the top-right corner.
3. Check if Microsoft Visual C++ 2010 x86 Redistributable – 10.0.40219 is listed in installed programs.
4. If it does not appear in the list, use Windows Update or download the package directly from Microsoft's website (select VSU_4\vcredist_x86.exe package from the download link).

4.1.3 Included files

The software tool and reference code for the qcaldc tool are provided under qcaldc_package_010101.TEMP file at the Attachments portion of this document, which can be accessed from any popular pdf viewer software. The files must be located in the same folder. Files supplied are:

- qcaldc.exe – Front-end executable; for sample code, see Section 4.3
- qcaldc.dll – Backend QTI proprietary calibration software library

- qcaldc.cfg – Sample configuration file
- Test vectors and reference outputs
  - main-L.bmp – Sample main sensor calibration image of the L-chart (3D chart).
  - main-Flat.bmp – Sample main sensor calibration image of the Flat-chart (2D chart)
  - aux-L.bmp – Sample auxiliary sensor calibration image of the L-chart (3D chart)
  - aux-Flat.bmp – Sample auxiliary sensor calibration image of the Flat-chart (2D chart)
  - qcaldc-reference.log – Output of qcaldc tool when run on the test vectors and configuration file

4.2 Usage

The qcaldc tool accepts two BMP image pairs, the main and auxiliary sensor image captures of the calibration L-chart (3D chart) and the Flat-chart (2D chart), respectively. For details regarding the calibration procedure, see Chapter 3.

4.2.1 Syntax

```
qcaldc.exe <configuration file>
```

It is advisable to redirect the output of the qcaldc tool to a log file for analysis or debugging.

Example usage

```
qcaldc qcaldc.cfg > qcaldc.log
```

4.2.1.1 Input parameters

The qcaldc.cfg configuration file requires the mandatory parameters described in Table 4-1.

Table 4-1 Mandatory parameters

Parameter name	Description
`main_native_width`	From main sensor specification – Full FOV, native sensor resolution (largest possible width)
`main_native_height`	From main sensor specification – Full FOV, native sensor resolution (largest possible height)
`main_f_number`	F number **Note**: This parameter is not currently used inside the calibration algorithm.
`main_fovd_degrees`	Diagonal FOV in degrees **Note**: This parameter is not currently used inside the calibration algorithm.
`main_pixel_pitch_mm`	From main sensor specification – Pixel pitch in mm
`main_focal_length_mm`	From main lens specification – Focal length at infinity in mm

Parameter name	Description
main_focal_length_ratio	Ratio of focal length during capture of calibration images to main_focal_length_mm
main_calib_input_width	Resolution of the calibration image that input to qcaldc (width) **Note**: Calibration algorithm expects this value to be same as main_native_width.
main_calib_input_height	Resolution of the calibration image that input to qcaldc (height) **Note**: Calibration algorithm expects this value to be same as main_native_height.
main_calib_L_chart_image	BMP filename of L-chart (3D chart) calibration image from main sensor
main_calib_flat_chart_image	BMP filename of Flat-chart (2D chart) calibration image from main sensor
aux_native_width	From auxiliary sensor specification – Full FOV, native sensor resolution (largest possible width)
aux_native_height	From auxiliary sensor specification – Full FOV, native sensor resolution (largest possible height)
aux_f_number	F number **Note**: This parameter is not currently used inside the calibration algorithm.
aux_fovd_degrees	Diagonal FOV in degrees **NOTE**: This parameter is not currently used inside the calibration algorithm.
aux_pixel_pitch_mm	From auxiliary sensor specification – Pixel pitch in mm
aux_focal_length_mm	From auxiliary lens specification – Focal length at infinity in mm
aux_focal_length_ratio	Ratio of focal length during capture of calibration images to aux_focal_length_mm
aux_calib_input_width	Resolution of the calibration image that is input to qcaldc (width) **Note**: Calibration algorithm expects this value to be same as aux_native_width.
aux_calib_input_height	Resolution of the calibration image that is input to qcaldc (height) **Note**: Calibration algorithm expects this value to be same as aux_native_height.
aux_calib_L_chart_image	BMP filename of L-chart (3D chart) calibration image from auxiliary sensor
aux_calib_flat_chart_image	BMP filename of Flat-chart (2D chart) calibration image from auxiliary sensor

Parameter name	Description
calib_square_size_mm	Size in mm of each side of the checkerboard pattern on the L-chart (3D chart)
cam_baseline_mm	Baseline separation between cameras in mm
main_aux_position_flag	▪ 0 – Main is on the left of auxiliary ▪ 1 – Main is on the right of auxiliary
L_chart_distance_mm	Physical distance in mm between module and L-chart (3D chart) during calibration
flat_chart_distance_mm	Physical distance in mm between module and flat-chart (2D chart) during calibration
calib_otp_data_file	Filename of binary dual camera calibration One-Time Programmable (OTP) data (output of qcaldc)
residual_vertical_disparity_threshold_pixels	Threshold for residual vertical disparity between main and auxiliary images after applying projective transform, in pixels at calibration resolution
scale_difference_threshold_percent	Threshold for scale difference of vertical lines in main and auxiliary images, in percentage
L_chart_origin_distance_threshold_percent	Threshold for L-chart (3D chart) origin distance error, in percentage

4.2.1.2 Additional explanation of mandatory input parameters

In Table 4-1 above, some parameters require special attention. Hence, additional explanation is provided below.

- main_focal_length_ratio should be calculated for the Main camera as follows:
  - For the module being calibrated, first record the VCM position (or DAC value) at the infinity position.
  - Next, follow instructions in Section 3.2.3 and set it to focus on the green square of the L- chart. Note that either Auto-focus or Manual focus can be used.
  - Record the VCM position (or DAC value) after focus is complete and the lens has settled. Capture the L-chart image at this lens position. Note that Flat-chart image should also be captured with the lens fixed at the same position.
  - Compute the lens shift (in mm) between infinity position and the current lens position. Then:

$$\text{main\_focal\_length\_ratio} = \frac{focal\_length\_at\_infinity\_mm + lens\_shift\_in\_mm}{focal\_length\_at\_infinity\_mm} \tag{4-1}$$

- aux_focal_length_ratio can be calculated in a similar way for the Auxiliary camera. If this is a fixed-focus camera, this ratio will be 1, *if* the lens has been fixed at the infinity position.
- main_aux_position_flag: In order to determine whether Main is on the left (0) or right (1) of the Auxiliary camera, the module should be positioned with respect to the calibration charts as shown in Figure 3-1. If the Main image captures more of the left field-of-view with respect to Auxiliary image, then Main is on the left (0) of Auxiliary camera. Otherwise, it is on the right (1) of Auxiliary camera.

4.2.2 Operation

The qcaldc tool prints status messages to the console during operation. This output is an example of the messages that qcaldc displays:

```
qcaldc version  = 0x10204
===========================
Validating config file qcaldc.cfg ...
===========================
Parsed config file qcaldc.cfg ...
===========================
Calibration verification results ...
                                            Metric  Value  Threshold Result
     L-chart: Residual vertical disparity [pixels]   1.52       3.00   PASS
                       L-chart: Distance error [%]   3.60      20.00   PASS
  Flat-chart: Residual vertical disparity [pixels]   1.86       3.00   PASS
                    Flat-chart: Distance error [%]   6.20      20.00   PASS
                  Flat-chart: Scale difference [%]   0.55       1.00   PASS

===========================
Writing calibration OTP data to qcaldc-otp.bin ...

===========================
qcaldc calibration success!
```

4.2.3 Output

If the input parameters are all present and correct, the qcaldc tool begins the calibration process. During the calibration, informational messages are printed on the screen. The qcaldc tool also prints verification metrics, along with thresholds and PASS/FAIL results.

- L-chart: Residual vertical disparity [in pixels] – Any remaining vertical misalignment between main and auxiliary L-chart images after applying a projective transform on the auxiliary image.
- L-chart: Distance error [%] – An error in the predicted distance to the center of the L- chart (3D chart).
- Flat-chart: Residual vertical disparity [in pixels] – Any remaining vertical misalignment between main and auxiliary Flat -chart images after applying a projective transform on the auxiliary image.
- Flat-chart: Distance error [%] – An error in the predicted distance to the center of the Flat- chart (2D chart).
- Scale difference [%] – Any scale difference between vertical lines in the main and auxiliary Flat-chart images after applying a projective transform.

If any one of the 5 validation metrics has FAILed, calibration has failed. In this case, qcaldc tool will not produce any binary output file. It is recommended that the FAILed module be pulled out from the production line for further inspection.

On the other hand, if all 5 validation metrics have PASSed, calibration has succeeded. In this case, qcaldc tool produces one binary output file. The name of the output file specified uses the `calib_otp_data_file` parameter in the qcaldc.cfg configuration file. The output file is a 470- byte binary file containing the calibration data. The calibration data format is shown in Table 4-2. Note that all the data fields are written out in little-endian byte order.

Table 4-2 Output file table

Start byte	Data	Number and type of values	Description
0	`version`	1*uint32_t	Calibration OTP format version
4	`main_focal_length_pixels`	1*float	Normalized focal length in pixels at calibration resolution; intrinsic parameter output by calibration
8	`main_native_width`	1*unsigned short	Native main sensor resolution used to capture calibration image (width)
10	`main_native_height`	1*unsigned short	Native main sensor resolution used to capture calibration image (height)
12	`main_calib_width`	1*unsigned short	Image size (width) used internally by qcaldc calibration tool
14	`main_calib_height`	1*unsigned short	Image size (height) used internally by qcaldc calibration tool
16	`main_focal_length_ratio`	1*float	Ratio of focal length of main during capture of calibration images to `main_focal_length_mm`
20	`aux_focal_length_pixels`	1*float	Normalized focal length in pixels at calibration resolution; intrinsic parameter output by calibration

Start byte	Data	Number and type of values	Description
24	aux_native_width	1*unsigned short	Native auxiliary sensor resolution used to capture calibration image (width)
26	aux_native_height	1*unsigned short	Native auxiliary sensor resolution used to capture calibration image (height)
28	aux_calib_width	1*unsigned short	Image size (width) used internally by qcaldc calibration tool
30	aux_calib_height	1*unsigned short	Image size (height) used internally by qcaldc calibration tool
32	aux_focal_length_ratio	1*float	Ratio of focal length of auxiliary during capture of calibration images to aux_focal_length_mm
36	R	9*float	Relative viewpoint matching matrix pertaining main
72	G	32*float	Relative geometric surface description parameters
200	ox	1*float	Relative offset of sensor center from optical axis along horizontal dimension
204	oy	1*float	Relative offset of sensor center from optical axis along vertical dimension
208	position	1*unsigned short	▪ 0 – Main camera is on the left of auxiliary ▪ 1 – Main camera is on the right of auxiliary
210	cam_baseline_mm	1*float	Camera separation baseline distance in mm
214	reserved	64*float	Reserved for future use

4.3 Sample code for qcaldc.exe front-end

The following is sample code for the qcaldc.exe front-end that can call into qcaldc.dll.

```
#include <windows.h>
#include <stdio.h>

#define QCALDC_FAILURE    (0)
#define QCALDC_SUCCESS    (1)

typedef int (*qcaldc_calibrate)(const char* qcaldc_config_filename);

int main(int argc, char *argv[])
{
  HINSTANCE          handle = NULL;
  qcaldc_calibrate func_ptr = NULL;

  handle = LoadLibrary(L"qcaldc.dll");
```

```
    if (!handle)
    {
      printf ("LoadLibrary of qcaldc.dll failed (err=%d)\n",GetLastError());
      return QCALDC_FAILURE;
    }

    func_ptr = (qcaldc_calibrate)GetProcAddress(handle,"qcaldc_calibrate");

    if (!func_ptr)
    {
      printf ("GetProcAddress failed (err=%d)\n",GetLastError());
      return QCALDC_FAILURE;
    }

    if (QCALDC_SUCCESS != func_ptr(argv[1]))
    {
      printf ("qcaldc calibration failed!\n");
      return QCALDC_FAILURE;
    }

    printf ("qcaldc calibration success!\n");

    if (handle)
      FreeLibrary(handle);

    return QCALDC_SUCCESS;
  }
```

A References

A.1 Related documents

Title	Number
Resources	
https://www.microsoft.com/en-us/download/details.aspx?id=30679	Microsoft

A.2 Acronyms and terms

Acronym or term	Definition
AF	Auto Focus
CFA	Color Filter Array
IAF	instantAF
FOV	Field of View
LCA	Logic Cell Array
NVM	Nonvolatile Memory
OTP	One-Time Programmable