Revision	ECO#	Description	Approved	Date
1	1797115	Eng Release	See ECO	Jan 20 2013
2	1890003	Eng Release	See ECO	Mar 19 2013
3	1932811	Eng Release	See ECO	Apr 4 2013
4	2083758	Eng Release	See ECO	May 1 2013
5	2503174	Eng Release	See ECO	Sept 19 2013
6	2944017	Eng Release	See ECO	May 14, 2014

Module Testing ERS for Auto Focus Cameras

Quarry ERS
Auto Focus Testing

Document 069-9867
Version 06
May 15, 2014

Editors:

Eugene Lam	lam_eugene@apple.com
Julianna Lin	julianna_lin@apple.com
Robin Jenkin	rjenkin@apple.com

Authors:

Camera Product Engineering Team

Table of Contents

About This Document

Specification Confidentiality

The contents of this Module Testing ERS, including any Apple-Vendor project-specific information, are Apple Confidential Information subject to the non-disclosure and use restrictions set forth in the Confidentiality Agreement between Vendor and Apple.

No part of this specification, algorithms, and its contents may be shared, distributed or disclosed, in any form, to any third party without explicit written agreement from the Apple DRI. This restriction includes the sharing of information by Vendor to its suppliers or customers without the prior written consent by Apple. Where possible, Apple will always try to directly distribute the specifications to the Vendor's customers or suppliers who need them as this is the preferred means of document distribution.

Additionally, requests from a Vendor's customers for information or for changes related to any specifications or process items should be referred to the appropriate Apple DRI for a response. However, the Vendor's customers can request, through the Apple DRI, that these specifications or process be audited by Apple personnel.

If any request is made by a third party contrary to the above requirements, such request should be immediately escalated to the appropriate Apple DRI for review and resolution. For the purposes of this specification, Vendor is defined as the company supplying the part/service specified by this document.

Audience

This specification assumes you have some familiarity with Apple portable products. The people who benefit from this guide are:

- Manufacturers who are involved in assembly and test of this image sensor module.
- Engineers who are designing components of this image sensor module.
- Engineers who are designing electrical or mechanical parts interfacing to or related to this image sensor module.

Authority

This specification provides direction on the setup and algorithms used for module testing. The direction in this document is superseded by those given in either the Module ERS or VSR.

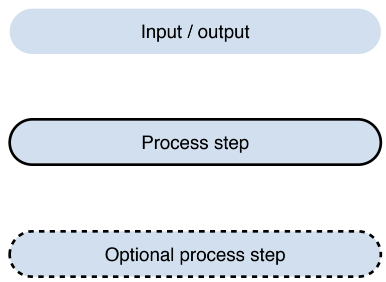
Reference

Note that all documentation assumes the use of the preprocess function to perform pre-processing steps on the image. Specific tests may opt to override certain aspects of this function to provide different functionality;text within specific test documentation supersedes all other text.

Conventions

bit-depth — Bit-depth corresponds to the bit-depth of the original image capture

flowcharts — Flowcharts use the following legend to denote inputs, outputs, and to differentiate between mandatory and optional process steps in the test methodology:



precision — Floating-point double precision, as defined by the IEEE Standard for Floating Point Arithmetic (IEEE 754), is assumed for all calculations, unless otherwise stated

resolution — Image resolution may be referred to as (*w* x *h*), with all references to *w* or *h* representing the width or height of the image, respectively

rounding — Rounding should follow the IEEE Standard for Floating Point Arithmetic (IEEE 754), using rounding towards the nearest integer value, with ties rounding away from 0

Definitions

Bayer image — The spatially subsampled R, Gr, Gb, and B colour channels (stored in this sequence) extracted from a RAW image, before using the demosaic algorithm to fill in the missing pixel values demosaicing. For image sensors that use a colour filter array, demosaicing is the algorithm for reconstructing a full colour image from the spatially undersampled R Gr Gb B Bayer channels in the RAW image

full scale — Corresponds to the bit-depth of the original image (e.g., for n-bits, full scale = 2^n - 1)

half scale — Half of the full scale deflection (e.g., for n-bits, half scale = 2^(n-1))

RAW image — Raw image data, as output from the module, consisting of a (w x h) array of integers in the same sequence as the pixels on the sensor chip, ignoring colour information.

RGB image — Image consisting of R, G, and B colour planes after demosaicing, and with the colour planes in that sequence

Y-channel	Y channel from YCbCr conversion
wBayer	The width of each Bayer channel after the RAW image is split. Also know as the x-direction of the image starting initial index on top left of image.
hBayer	The height of each Bayer channel after the RAW image is split. Also know as the y-direction of the image starting initial index on top left of image.

Below is the Apple reference SFR chart with the AWB exposure target marked. This figure is to be referenced for any test that involves AWB and the SFR chart.

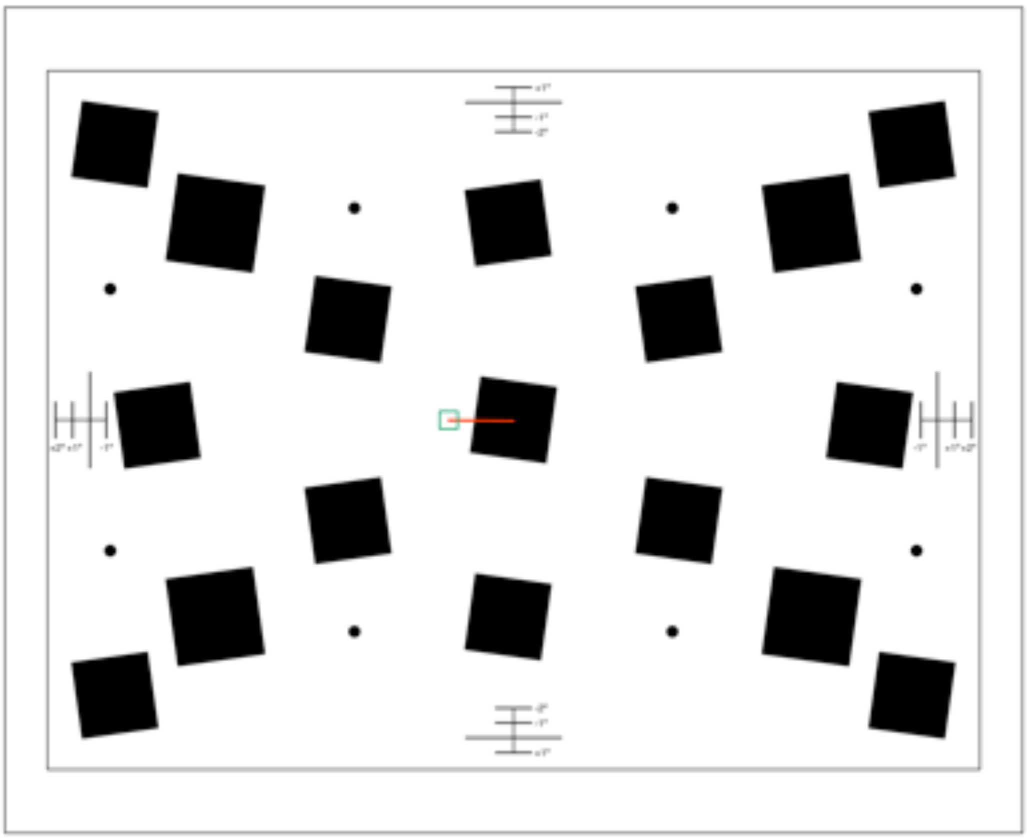


Figure 1: Auto-exposure window (green)

Log File

Test logs shall be saved for every test run on every module. Each log shall adhere to the following guidelines:

- comma-separated .txt or .csv file
- first row contains column headers (case-sensitive column header naming is provided in this document)
- subsequent rows contain one set of module data per row
- module data traced with a “barcode” column containing module barcode
- sensor ID traced with a “sensorID” column containing sensor ID
- test time and lot number with each module tested
- tester and socket with each module tested
- additional columns may be added, but column headers must be unique.

Test logs are to be made available to Apple Engineering upon request.

Raw, unprocessed images should be saved for each module for each test configuration up to and including the final C2.1 build and be made available upon request.

After C2.1, 20k material per config should be uploaded to Radar on a weekly basis for review.

Preprocessing

This document outlines any pre-processing steps that may be done on an image prior to input for image analysis.

Methodology

The input image consists of a (w x h) array of integers in the same sequence as the pixels on the sensor chip, ignoring colour information (see Figure 2).

(1,1)	(2,1)	(3,1)	(4,1)		
(1,2)	(2,2)	(3,2)	(4,2)		
(1,3)	(2,3)	(3,3)	(4,3)		
(1,4)	(2,4)	(3,4)	(4,4)		

Figure 2: Indexing methodology for RAW images

Any frame averaging should be done prior to input to the preprocess algorithm and the values rounded to integers before any additional processing. The final output of the preprocess algorithm will vary depending on the desired format, but will always be in integer format.

Of particular note are the sequence of the pre-processing steps. Certain tests may choose to omit certain processing steps, in which case these may be bypassed, but the sequence should remain the same.

1. Subtract the pedestal value, according to the value specified in the ERS (e.g. 32 LSB). The output of this step should be a signed value.

2. Apply a grey-world white balance algorithm to the image to bring all the colour channels to the same value in the centre. (Note: Certain tests, such as SFR, will require a different region of interest, and those instructions should supersede the instructions provided below.)

   2.1. Calculate the average value for the 100x100 centre pixels of each colour channel, and name the values:

      cenMean_R
      cenMean_Gr
      cenMean_Gb
      cenMean_B

2.2. Calculate the correction factors corrWB_R, corrWB_Gr, corrWB_Gb, corrWB_B for each colour channel as follows:

maxCenMean = maximum value from the means (*cenMean_R*, *cenMean_Gr*, *cenMean_Gb*, *cenMean_B*)
corrWB_R = *maxCenMean* / *cenMean_R*
corrWB_Gr = *maxCenMean* / *cenMean_Gr*
corrWB_Gb = *maxCenMean* / *cenMean_Gb*
corrWB_B = *maxCenMean* / *cenMean_B*

2.3. Apply the correction factor to each colour channel respectively by multiplying each pixel by the corresponding correction factor.

2.4. Clip any values to ensure that no pixel is above the bit-depth saturation level, e.g. for a 10-bit image, values should be clipped to 1023.

3. Apply software lens shading:

3.1. Using the index values (x, y) from the original RAW image (see Figure 1), calculate the following variables:

centreX = w / 2 + 0.5
centreY = h / 2 + 0.5

$$halfDiag = \sqrt{(centreX - 1)^2 + (centreY - 1)^2}$$

$$pixDistance(x,y) = \sqrt{(x - centreX)^2 + (y - centreY)^2}$$

$$FOV\_pixel(x,y) = \frac{FOV}{2} \cdot \frac{pixDistance}{halfDiag}$$

where FOV is the value in degrees provided within the ERS

3.2. For each pixel, apply the following gain

$$pixNew(x,y) = \frac{1}{\cos(FOVpixel(x,y))^4}$$

3.3. Clip any values to ensure that no pixel is above the bit-depth saturation level, e.g. for a 10-bit image, values should be clipped to 1023.

4. Apply a bilinear demosaicing algorithm to all pixels. For a given pixel:

   4.1. If it is the original pixel colour, use the original value.

   4.2. For all other colours, average all directly adjacent pixels for that colour.

   This will allow for 1, 2, and 3 pixel averaging on the edges; 2 and 4 pixel averaging in the centre. The image below outlines the number of pixels available for averaging on each colour plane:

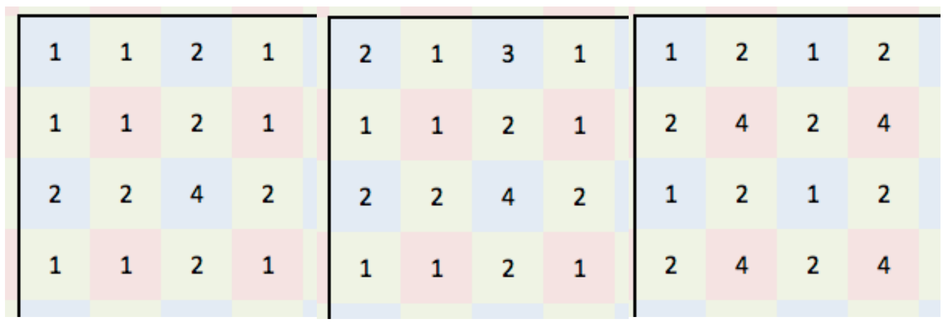


Figure 3: Diagram showing the number of pixels used for averaging at each pixel for the R (left), G (middle), and B (right) channels.

5. Convert from RGB to YCbCr using the following equations:

   Y = 0.299R + 0.587G + 0.114B
   Cb = half full scale - 0.1687R - 0.3313G + 0.5B
   Cr = half full scale + 0.5R + 0.4187G - 0.0813B

6. Round the Y-channel to integer values.

Methodology Flowchart

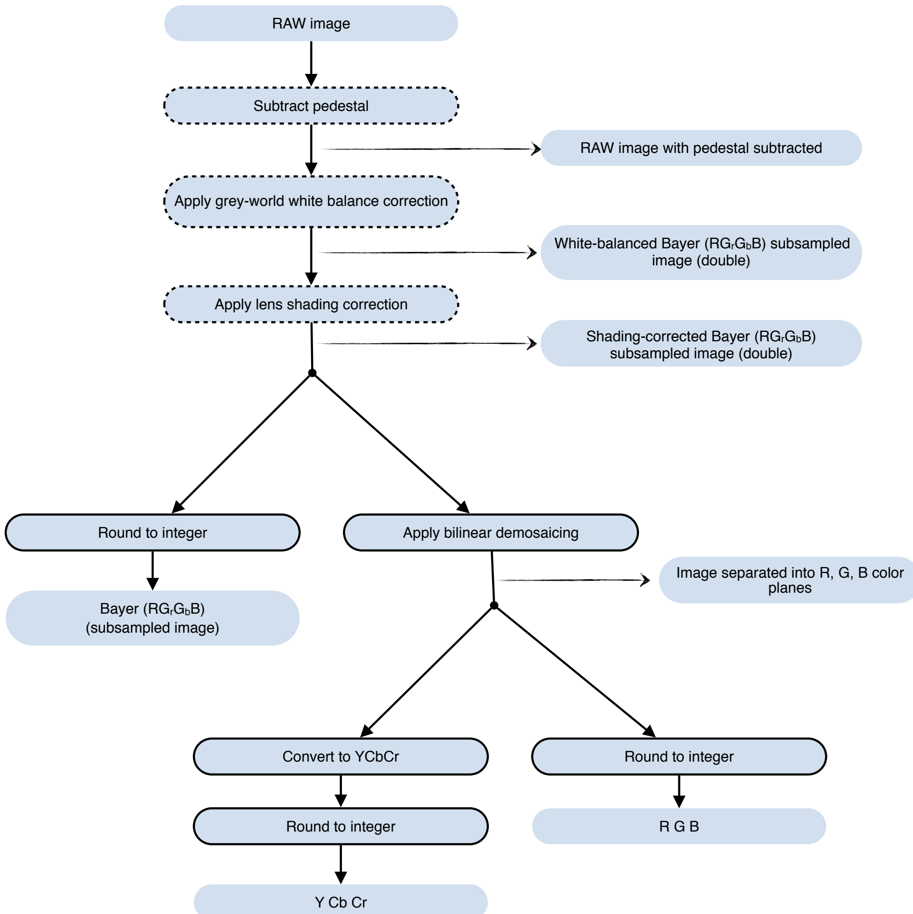


Figure 4: Preprocessing Methodology Flowchart

Autofocus Calibration

Autofocus calibration is used to determine the best lens focus positions at both macro and infinity distance (where "infinity" refers to some far distance. These positions are programmed into the NVM for further use downstream. Correct calibration has a direct impact to the ability of the camera to perform, and also provides additional process feedback.

Test Hardware Setup

A transmissive Apple SFR chart with D50 light source should be used as the test target.

Parameter	Macro	Infinity
Illuminant	D65	D65
Light Level	>200 cd/m	>200 cd/m
Target	high contrast Transmissive Apple SFR test target	high contrast Transmissive Apple SFR test target
Camera distance to target	Specific distances will be set by the camera module ERS. Use the same distances as those specified for measuring SFR. Setup distances must be measured with a calibrated laser distance meter	Specific distances will be set by the camera module ERS. Use the same distances as those specified for measuring SFR. Setup distances must be measured with a calibrated laser distance meter
Orientation	module face up	module face down

Test Chart

Please refer to the documentation on SFR charts for more details.

Module Hardware Setup

Module setup should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Software driven AE	70±5% FSD	average of a 64x64 pixel ROI around the white area which is left of the centre tilted square by 10% of the image width (refer to Figure 1) post pedestal subtraction, on y-channel
Analog gain	1x	
Integration time	<66.67 ms	Controlled by SW exposure adjust, limited by frame rate
DPC	off	
Frame rate	15 fps	Frame rate limits max exposure time to 1/15 sec = 66.7msec
Sensor LSC	disabled	
Sensor AWB	disabled	

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). It is not necessary to save all images - only the final focus position image shall be saved.

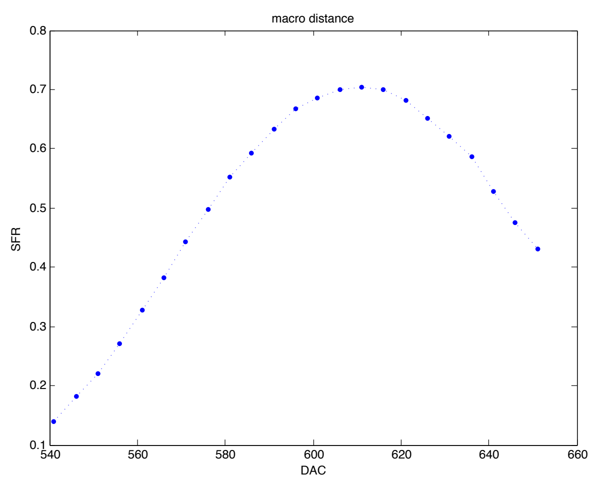
Test Image Specification	Value
Number of image captures	1 image capture.
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

The autofocus calibration procedure is very program dependent and is only covered on a high-level in this documentation. Specific tuning with Apple engineers is required for this procedure. The general procedure is outlined below:

For macro distance

1. Use the actuator to sweep through different lens positions, calculating the centre SFR value at Ny/4 for each sweep image. It is acceptable to reduce the sweep range to reduce the test time.

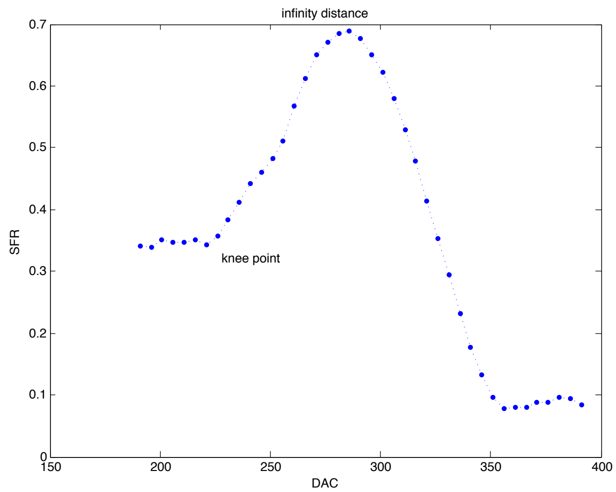


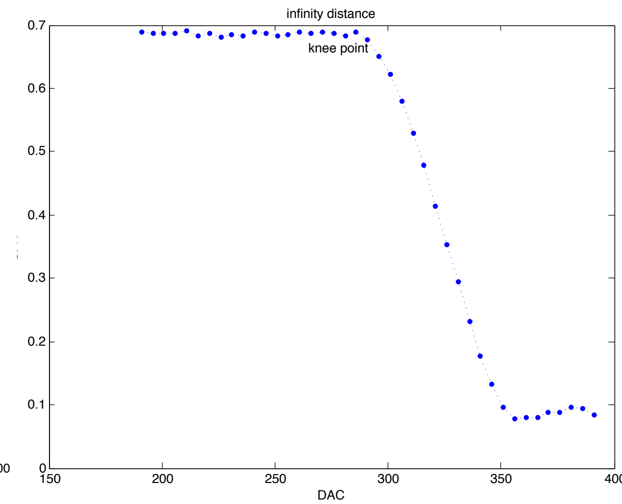
2. Detect the peak of the sweep and add one step to this value. This is the value that will be programmed into the NVM. Note that in addition, the following flag should exist to indicate the presence or lack of presence of a peak according to the following:

```
if max(SFR) > SFR(firstPos) + offset
    sfr_X_peakFlag = -1                     % peak is on the left limit
elseif max(SFR) < SFR(lastPost) + offset
    sfr_X_peakFlag = 1;                     % peak is on the right limit
else
    sfr_X_peakFlag = 0;                     % peak exists
end
```

For infinity distance

1. Use the actuator to sweep through different lens positions, calculating the centre SFR value at Ny/4 for each sweep image. It is acceptable to reduce the sweep range to reduce the test time.





2. Detect the peak of the sweep and add one step to this value. Note that in addition, the following flag should exist to indicate the presence or lack of presence of a peak according to the following:

```
if max(SFR) > SFR(firstPos) + offset
    sfr_X_peakFlag = -1;              % peak is on the left limit
elseif max(SFR) < SFR(lastPos) + offset
    sfr_X_peakFlag = 1;               % peak is on the right limit
else
    sfr_X_peakFlag = 0;               % peak exists
end
```

3. Detect the “knee” of the sweep. The value programmed into the NVM should be done according to the following:

```
if sfr_X_peakFlag == -1
    NVM_programValue = kneePos - 32;  % no peak, go back left for margin
else
    NVM_programValue = kneePos;       % peak exists
end
```

Results Naming Convention in Log File

Name	Description	Unit
sfr_X_peakFlag	peak detected	code

Colour Calibration

Colour calibration for the mass production of camera modules should follow the design and specifications described in this section. The aim is to provide a common method and light source design to achieve consistent colour calibration results across integrators and reduce the probability of errors during production. Data stored in NVM are the actual colour ratios recorded directly from the DUT module, not a quotient relative to a reference module measurement. For example, if the DUT module reads 0.80 for R/G under A light, then that value is converted to a hex value and stored in NVM.

Colour Calibration Setup

The light box provides a fixed stimulus (fixed colour temperature, controlled spectral distribution and uniform illumination) to the camera module under test. Images from the module are analyzed by the computer in the test station to obtain the raw colour ratios that are then programmed into the first NVM block. The second NVM block is reserved for re-work. The exact methods used to compute the NVM values are described in the section, “Colour Ratio Calculations.”

Test Hardware Setup

An “A light” source and a “D50 light” source fitted with an opal diffuser (as per section: Camera Calibration Light Sources) should be used as the test targets. The light sources must be warmed up for 5 minutes after each time they are powered off. The 5-minute warmup is not required after daily light source system calibration (using the i1Pro2) unless the light source is powered off.

	A Light	D50 Light
Illuminant	Halogen	Filtered Halogen
Colour temperature	2800K ± 20K	5000K ± 20K
Light level	300-800 cd/m	100-300 cd/m
Target	Opal Diffuser	Opal Diffuser
Distance to target	8mm - 16mm	8mm - 16mm
Orientation	module face up	module face up

Module Hardware Setup

Five RAW image captures are to be averaged and saved as a single averaged RAW image.

Configuration	Value	Comments
lens position	focused @ 2m	Lens position must be in a deterministic position.
SW Exposure Adjust	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Image resolution	full	
Bit depth	full	Typical 10-bit ADC output
Pedestal Subtract	yes	Verify level with specific camera module ERS
Analog Gain	1x	
Digital Gain	1x	
Integration Time	<66.67 ms	Controlled by SW exposure adjust, limited by frame rate
DPC	on	Refer to specific camera module ERS
Frame Rate	15 fps	
Colour Shading	off	
AWB	off	

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). Five RAW image captures are to be averaged and saved as a RAW image.

Test Image Specification	Value
Number of image captures	5 RAW frames, which should be averaged
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

To calculate the raw colour ratios R/G and B/G for a particular module, apply the following procedure using both light sources (A and D50):

1. Log camera module serial number

2. Average two frames in uncompressed raw Bayer format

3. Separate into Bayer channels: R, Gr, Gb, B.

4. Extract the 50x50 pixels at image centre from each Bayer channel

5. Compute the average pixel value for each Bayer channel, call them R_mean, Gr_mean, Gb_mean, B_mean. Compute the average of Gr_mean and Gb_mean, call it G_mean.
   - For A light, log these values under columns "*cc_A_R*", "*cc_A_G*" and "*cc_A_* B".
   - For D50 light, log these values under columns "*cc_D50_R*", "*cc_D50_G*" and "*cc_D50_* B".
   - Record data with at least two decimal places of precision (e.g. "%6.2f" or larger format).

6. Compute the colour ratios R/G and B/G as (R_mean / G_mean) and (B_mean / G_mean).
   - For A light, log these values under columns "*cc_A_RG*" and "*cc_A_BG*".
   - For D50 light, log these values under columns "*cc_D50_RG*" and "*cc_D50_BG*".
   - Record data with at least four decimal places of precision (e.g. "%7.4f" or larger format).

7. During light source calibration, Residual Error Correction (REC) values were calculated. The valid range of these REC values is [0.97 to 1.03]. If any REC value falls outside this range, notify Apple, change the light bulb, and re-calibrate with the new bulb.

8. Multiply the floating point colour ratios by the REC values.
   - For A light, log these values under columns "*cc_A_RG_cor*" and "*cc_A_BG_cor*".
   - For D50 light, log these values under columns "*cc_D50_RG_cor*" and "*cc_D50_BG_cor*".
   The data should be recorded with at least four decimal places of precision (e.g. "%7.4f" or larger format).

A light Example:

> Raw Image after pedestal subtract: cc_A_R=800, cc_A_G=1000, cc_A_B=300
> Colour ratios: cc_A_RG = 0.800, cc_A_BG = 0.300
> REC factors reported by software: cc_A_RG_REC = 1.01, cc_A_BG_REC = 0.99
> Corrected ratios: cc_A_RG_cor = 1.01 * 0.800 = 0.808,
> cc_A_BG_cor = 1.01 * 0.300 = 0.303

9. The valid range of colour ratios is [0.0 to 2.0].

10. Convert each colour ratio to an unsigned integer: multiply by a scaling factor, and then round to the nearest integer. The scaling factor is listed in the specific camera module program ERS.

11. Convert the unsigned integer colour ratios to hexadecimal and write to NVM. The bit depth and NVM locations are listed in the specific camera module program ERS.

12. For A light, log these values in hexadecimal format under columns "cc_A_RG_NVM" and "cc_A_BG_NVM".
    For D50 light, log these values in hexadecimal format under columns "cc_D50_RG_NVM" and "cc_D50_BG_NVM".
    The data should be recorded in "0x%4x" format.

Reference and Limit Module Selection

A Reference Module is one that has been selected by the “Camera Calibration Modules.xls” spreadsheet based on the measured and reported colour ratios. A separate set of Reference Modules must be collected for each module configuration with a different image sensor and lens combination. Variations in components outside the optical path (e.g. Actuator, substrate, flex) are not critical from the standpoint of colour calibration. New Reference Modules are only selected during engineering builds, unless otherwise directed by Apple.

Use the following procedure to select colour calibration Reference and Limit Modules for shipment to Apple.

1.Build 300 working camera modules, and measure the R,G,B colour responses under both the “A” and “D50” illuminants.

2.Open the provided Excel spreadsheet “Colour Calibration Modules.xls”.

3.On the first worksheet tab named “Input data”, enter the following data for each module:

Module ID	Sensor ID

cc_A_R	cc_A_G	cc_A_B	cc_A_RG	cc_A_BG	cc_A_RG_cor	cc_A_BG_cor	cc_A_RG_NVM	cc_A_BG_NVM
cc_D50_R	cc_D50_G	cc_D50_B	cc_D50_RG	cc_D50_BG	cc_D50_RG_cor	cc_D50_BG_cor	cc_D50_RG_NVM	cc_D50_BG_NVM

4.Push the button named: Run

5.On the third worksheet tab named “Output results” note the new column:
Type = Reference-Apple, Reference-Keep, Limit #1, Limit #2, or blank

6.Set aside the Reference-Keep modules and store in a safe place. Apple may require the use of those modules in the future.

7.Put the Reference-Apple and all Limit modules into 3 trays as shown in the figure below.

7.1.In Tray 0, Reference-Apple module 1 goes into cell A2, Reference-Apple module 2 goes into cell A4, etc...

7.2.In Tray 1, place the Limit Set #1 modules across the top row beginning with A1, and no spaces.

7.3.In Tray 2, place the Limit Set #2 modules in the tray edges according to the figure. Combine the A and D50 modules for this purpose. For example Min R/G can be minimum under A or D50.

8.Placement of Modules in Trays to be sent to Apple

8.1.Construct a spreadsheet named “Module Tray Position.xls” to serve as an index for the modules placed in the trays. For example, Sheet 1 should index Tray 0: Reference Modules. The cell A2 should contain the module ID of the module inside tray location A2. That module should be Reference-Apple Module 1 from the module colour ratio spreadsheet.

9.Send the 3 trays to Apple.

10.Email both spreadsheets (Colour Calibration Modules.xls and Module Tray Position.xls) to Apple.

Results Naming Convention in Log File

Name	Description	Unit
cc_A_R	Colour Calibration A-Light Red value	LSB
cc_A_G	Colour Calibration A-Light Green value	LSB
cc_A_B	Colour Calibration A-Light Blue value	LSB
cc_A_RG	Colour Calibration A-Light Red/Green ratio	%
cc_A_BG	Colour Calibration A-Light Blue/Green ratio	%
cc_A_RG_rec	Colour Calibration A-Light Red/Green correction factor	% (of 1)
cc_A_BG_rec	Colour Calibration A-Light Blue/Green correction factor	% (of 1)
cc_A_RG_cor	Colour Calibration A-Light Red/Green REC-corrected ratio	% (of 1)
cc_A_BG_cor	Colour Calibration A-Light Blue/Green REC-corrected ratio	% (of 1)
cc_A_RG_NVM	Colour Calibration A-Light Red/Green NVM value (hexadecimal)	none
cc_A_BG_NVM	Colour Calibration A-Light Blue/Green NVM value (hexadecimal)	none
cc_D50_R	Colour Calibration D50-Light Red value	LSB
cc_D50_G	Colour Calibration D50-Light Green value	LSB
cc_D50_B	Colour Calibration D50-Light Blue value	LSB
cc_D50_RG	Colour Calibration D50-Light Red/Green ratio	%
cc_D50_BG	Colour Calibration D50-Light Blue/Green ratio	%
cc_D50_RG_REC	Colour Calibration A-Light Red/Green correction factor	%
cc_D50_BG_REC	Colour Calibration A-Light Blue/Green correction factor	%
cc_D50_RG_cor	Colour Calibration D50-Light Red/Green REC-corrected ratio	%
cc_D50_BG_cor	Colour Calibration D50-Light Blue/Green REC-corrected ratio	%
cc_D50_RG_NVM	Colour Calibration D50-Light Red/Green NVM value (hexadecimal)	none
cc_D50_BG_NVM	Colour Calibration D50-Light Blue/Green NVM value (hexadecimal)	none

Colour Shading

Colour shading calibration corrects for lens shading, low-spatial-frequency sensor imperfections, and lens-sensor mismatch errors. It performs and stores a grid-based sampling of a uniform flat-field image. These sampled values are used by the device image signal processing (ISP) software to perform image adjustments.

Colour Shading Calibration Procedure

Per module Colour Shading values are to be programmed in the NVM in the locations indicated in the NVM map. The specific camera ERS will list the following:

- NVM map (any version in the VSR will supersede the ERS)
- wGrid, hGrid: dimensions of Regions Of Interest (ROI's) grid
- Array of Region Of Interest (ROI) locations

Test Hardware Setup

A "D50 light" source fitted with an opal diffuser (as per section: Camera Calibration Light Sources) should be used as the test target.

	D50 Light
Illuminant	Filtered Halogen
Colour temperature	5000K ± 20K
Light level	100-300 cd/m
Target	Opal Diffuser
Distance to target	8mm - 16mm
Orientation	module face up

Module Hardware Setup

Configuration	Value	Comments
Lens position	focused @ 2m	Lens position must be in a deterministic position.
SW Exposure Adjust	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog Gain	1x	
Integration Time	<66.67 ms	Controlled by SW exposure adjust, limited by frame rate
DPC	on	Refer to specific camera module ERS
Frame Rate	15 fps	
LSC	off	
White Balance	off	Off at sensor, correction will be manually applied

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). Five RAW image captures are to be averaged and saved as a RAW image.

Test Image Specification	Value
Number of image captures	5 RAW frames, which should be averaged
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

Use the D50 colour calibration system averaged RAW image to obtain the shading values as follows:

1. Average five frames in uncompressed raw Bayer format.
2. Preprocess the images (see "preprocess" document for more details): subtract pedestal from each image, and separate the data into its four Bayer channels. Do not perform any of the other pre-processing steps. Bayer channel data should be used for all subsequent calculations.

For each Bayer channel extract a wGrid x hGrid array of 10pixel x 10pixel

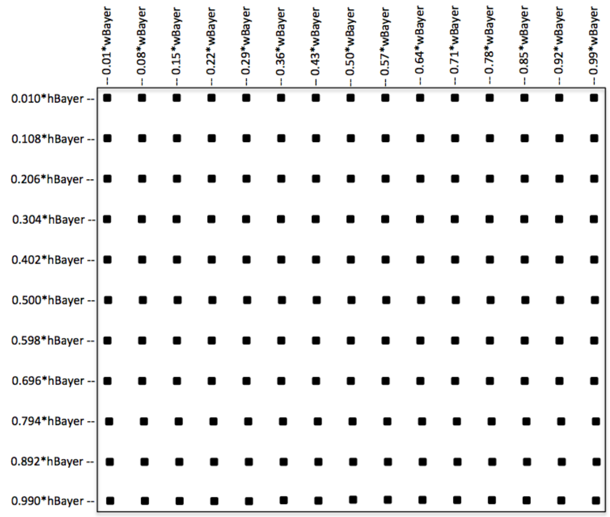


Figure 5: Example of grid pattern for colour shading with wGrid = 15, hGrid = 11

3. Compute the median value of each 10x10pixel area to give four wGrid x hGrid grids: R(i,j), GR(i,j), GB(i,j), B(i,j). The 10x10 pixel ROI's should be calculated as per below:

$$coordROI_{xLeft(i)} = round\left(round\left(roiX_i\right) - \frac{roiSize}{2} + 1.4999 \right)$$
$$coordROI_{yTop(j)} = round\left(round\left(roiY_j\right) - \frac{roiSize}{2} + 1.4999 \right)$$
$$coordROI_{xRight(i)} = round\left(round\left(roiX_i\right) - \frac{roiSize}{2} - 0.5 \right)$$
$$coordROI_{yBottom(j)} = round\left(round\left(roiY_j\right) - \frac{roiSize}{2} - 0.5 \right)$$

where

roiXi is the x-grid location for (i, j)
roiYj is the y-grid location for (i, j)
roiSize = 10

4. Scale each grid as follows:

scaled_GR(i,j) = GR(i,j) * 250.0 / max (GR(i,j))
scaled_R(i,j) = R(i,j) * 250.0 / max (R(i,j))
scaled_B(i,j) = B(i,j) * 250.0 / max (B(i,j))
scaled_GB(i,j) = GB(i,j) * 250.0 / max (GB(i,j))

5. For each scaled grid point, round to the nearest integer and program into the NVM as specified in the NVM map.

Test Methodology Flowchart

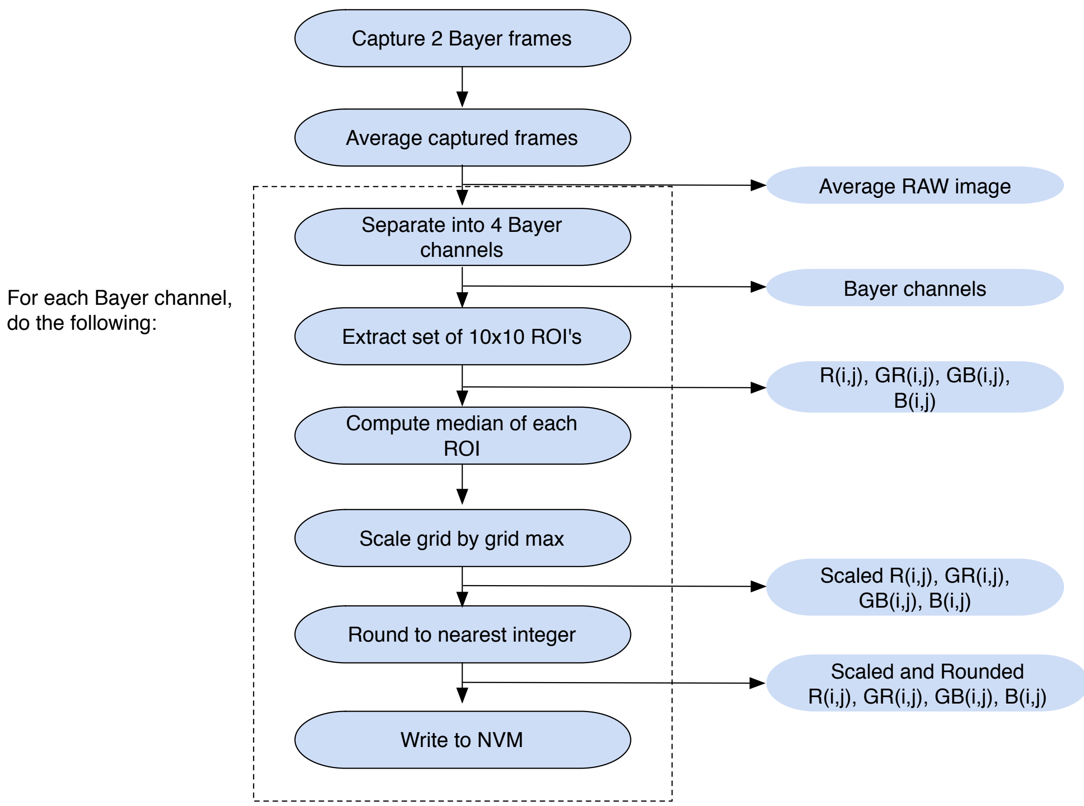


Figure 6 : Colour Shading Methodology Flowchart

Results Naming Convention in Log File

Name	Description	Unit
cs_R_gridScaled_x_y	Scaled and Rounded Red grid points, where x=horizontal, y=vertical indices	none
cs_Gr_gridScaled_x_y	Scaled and Rounded Green(Red) grid points, where x=horizontal, y=vertical indices	none
cs_Gb_gridScaled_x_y	Scaled and Rounded Green(Blue) grid points, where x=horizontal, y=vertical indices	none
cs_B_gridScaled_x_y	Scaled and Rounded Blue grid points, where x=horizontal, y=vertical indices	none

Colour Uniformity

This document defines the colour uniformity (CU) test setup and test methodology to be used for screening out colour non-uniformity caused by: 1) abnormal colour shading behaviour, and 2) diffused colour blemish caused by defects in the image sensor response or the IR-cut filter transmittance.

Colour Uniformity Setup

Test Hardware Setup

A D50 light source fitted with an opal diffuser (as per Appendix Appendix A) should be used as the test target.

Parameter	Value
Illuminant	D50
Correlated colour temperature	5000K ± 20K
Light level	100-300 cd/m
Target	Opal diffuser
Camera distance to target	8 - 16mm
Orientation	module up

Module Hardware Setup

Configuration	Value	Comments
Lens position	focused @ 2m	Lens position must be in a deterministic position.
SW Exposure Adjust	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog Gain	1x	
Integration Time	<66.67 ms	Controlled by SW exposure adjust, limited by frame rate
DPC	on	Refer to specific camera module ERS
Frame Rate	15 fps	
LSC	off	manual LSC will be applied on test images
White Balance	off	manual WB will be applied on test images

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). Five RAW image captures are to be averaged and saved as a RAW image.

Test Image Specification	Value
Number of image captures	5 RAW frames, which should be averaged
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

Below are the image processing steps to derive the CU scores.

1. Average 5 raw images captured under the test conditions outlined in the Test Hardware Setup and Module Hardware Setup tables.
2. Preprocess the image (see “preprocess” document for more details):
    2.1. Subtract pedestal from image, according to the value specified in the ERS (e.g. 16 LSB).
    2.2. Separate the image data into the four Bayer channels (i.e., R, Gr, Gb, B).
    2.3. Apply a grey-world white balance algorithm to the image to bring all the colour channels to the same value in the centre (see “preprocess” document for more details).
    2.4. Recombine the Bayer channels into a single array, using the original order of pixels from the original sensor image.
    2.5. Separate the image data into the four Bayer channels (i.e., R, Gr, Gb, B). Bayer channel data should be used for all subsequent calculations.
3. Calculate the colour shading NVM coefficient map based on the ColourShading test.
4. Replicate left, right, top, and bottom elements of colour shading map for each Bayer channel to account for border values of the image not sampled, divide map by 250 to normalize. Add an element with the value 0 to the beginning of the x locations of the colour shading map and an element with the value 1 to the end, increasing the number of elements by 2, repeat for the y locations.
5. For each Bayer channel, perform 2D bilinear interpolation using the colour shading map to obtain per pixel colour shading.

6. Apply software lens shading correction using per pixel colour shading.

    6.1. For each pixel, apply the following gain

    LSCgain = 1/perPixelLS

    where perPixelLS is the per pixel colour shading map obtained in step 5.

    6.2. Clip any values to ensure that no pixel is above the bit-depth saturation level, e.g. for a 10-bit image, values should be clipped to 1023.

7. Divide each Bayer channel into multiple ROIs, where the appropriate ROI size is defined by the module ERS (e.g., 9-by-9 Bayer pixels), and average the pixels in each ROI to form a binned version of the Bayer channel. In the case where the array size is not evenly divisible by the size of the ROI, the residual rows and columns should be evenly assigned to the border ROIs, resulting in border ROIs that may have a few more rows and/or columns than the other ROIs.

8. Calculate the colour parameters a and b for each ROI:

    8.1. *a* is the difference between the G and R channels, normalized by G, i.e.,

$$a = (G - R)/G$$

    8.2. *b* is the difference between the G and B channels, normalized by G, i.e.,

$$b = (G - B)/G$$

    where

$$G = (G_r + G_b)/2$$

9. Calculate the chroma of each ROI, which is defined as the Euclidean distance of the a and b colour parameters.

$$Chroma = \sqrt{(a^2 + b^2)}$$

10. Report the maximum chroma value, scaled by 100.

11. Again, find the ROI with the maximum Chroma value and calculate the colour difference ΔE of each ROI relative to the a and b values for the maximum Chroma ROI.

$$\Delta E(i,j) = \sqrt{\left(a(i,j) - a(\max Chroma)\right)^2 + \left(b(i,j) - b(\max Chroma)\right)^2}$$

12. Find the maximum ΔE value, scale it by 100, and call this value CU_maxDeltaE

$$CU\_\max DeltaE = \max\left(\Delta E(i,j)\right) \times 100$$

For each ROI, calculate the local colour difference ΔE(i,j,n) between the ROI under test and each of its 8 surrounding neighbours, as shown below:

Neighbor 1 (i-1, j-1)	Neighbor 2 (i-1, j)	Neighbor 3 (i-1, j+1)
Neighbor 4 (i, j-1)	ROI_underTest (i, j)	Neighbor 5 (i, j+1)
Neighbor 6 (i+1, j-1)	Neighbor 7 (i+1, j)	Neighbor 8 (i+1, j+1)

$$\Delta E(i,j,n) = \sqrt{\left(a(i,j) - a(Neighbor\_n)\right)^2 + \left(b(i,j) - b(Neighbor\_n)\right)^2}$$

13. Find the maximum local colour difference value for each ROI, and store this in a new array, called localDeltaE_map(i, j).
14. Scale localDeltaE_map by a factor of 100, so that it is in percentage.
15. Divide the localDeltaE_map into centre, edge, and corner regions, using the corner size specified in the module ERS (e.g., 3x3 blocks).
16. Calculate the following three metrics:

cu_localDE_cen_max	the maximum local DeltaE of the centre region.
cu_localDE_edg_max	the maximum local DeltaE of the edge region.
cu_localDE_corner_max	the maximum local DeltaE of the corner region.

17. Record the (x,y) coordinates for cu_chroma_max, cu_deltaE_max, cu_localDE_cen_max, cu_localDE_edg_max, and cu_localDE_corner_max, where x and y are the coordinates of the centre pixel of the ROIs, using the co-ordinate system from the original full-resolution image.

Test Methodology Flow Chart

The algorithm flow chart is illustrated below.

Figure 7: *colour uniformity test algorithm flow chart.*

Results Naming Convention in Log File

Note: “*” is to be replaced by the frequency setting per data. (i.e. “hf” or “lf)

Name	Description	Unit
cu_*_chroma_max	max Chroma value	
cu_*_deltaE_max	max colour difference DeltaE	%
cu_*_localDE_cen_max	max local DeltaE value for centre region	%
cu_*_localDE_edg_max	max local DeltaE value for edge region	%
cu_*_localDE_corner_max	max local DeltaE value for corner region	%
cu_*_chroma_max_x	the x coordinate (indexed back to the full-resolution image) corresponding to cu_chroma_max	pixel
cu_*_chroma_max_y	the y coordinate (indexed back to the full-resolution image) corresponding to cu_chroma_max	pixel
cu_*_deltaE_max_x	the x coordinate (indexed back to the full-resolution image) corresponding to cu_deltaE_max	pixel
cu_*_deltaE_max _y	the y coordinate (indexed back to the full-resolution image) corresponding to cu_deltaE_max	pixel
cu_*_localDE_cen_max_x	the x coordinate (indexed back to the full-resolution image) corresponding to max local_DE for centre region	pixel
cu_*_localDE_cen_max_y	the y coordinate (indexed back to the full-resolution image) corresponding to max local DeltaE for centre region	pixel
cu_*_localDE_edge_max_x	the x coordinate (indexed back to the full-resolution image) corresponding to max local DeltaE for edge region	pixel
cu_*_localDE_edge_max_y	the y coordinate (indexed back to the full-resolution image) corresponding to max local DeltaE for edge region	pixel
cu_*_localDE_corner_max _x	the x coordinate (indexed back to the full-resolution image) corresponding to max local DeltaE for corner region	pixel
cu_*_localDE_corner_max _y	the y coordinate (indexed back to the full-resolution image) corresponding to max local DeltaE for corner region	pixel

Sample Algorithm Parameter Settings

Parameter	Example Setting 1 (High Frequency)	Example Setting 2 (Low Frequency)
ROI Size (row x column pixels)	9 x 9	30 x 30
Corner Region (row x column blocks)	10 x 10	3 x 3

Contrast

The Contrast Test is a measure of image "misting" or low contrast defects, such as those caused by lens contamination or scratches.

Test Hardware Setup

A transmissive Apple SFR chart with D50 light source should be used as the test target.

Parameter	Value
Illuminant	D65
Light Level	>200 cd/m
Target	high contrast Transmissive Apple SFR test target
Camera distance to target	Specific distances will be set by the camera module ERS. Use the same distances as those specified for measuring SFR. Setup distances must be measured with a calibrated laser distance meter
Orientation	Any orientation is acceptable

Test Chart

Please refer to the documentation on SFR charts for more details.

Module Hardware Setup

Module setup should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Software driven AE	70±5% FSD	average of a 64x64 pixel ROI around the white area which is left of the centre tilted square by 10% of the image width (refer to Figure 1) post pedestal subtraction, on y-channel
Analog gain	1x	
Integration time	<66.67 ms	Controlled by SW exposure adjust, limited by frame rate
DPC	off	
Frame rate	15 fps	Frame rate limits max exposure time to 1/15 sec = 66.7msec
Sensor LSC	disabled	manual LSC will be applied on test images
Sensor AWB	disabled	manual WB will be applied on test images

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). One RAW image capture is to be saved.

Test Image Specification	Value
Number of image captures	1 image capture.
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

1.Capture 1 RAW image using the test conditions outlined above.

2.Preprocess the image (see “preprocess” document for more details): subtract pedestal from image, separate data into four Bayer channels, apply AWB, apply LSC, and convert to YCrCb space. Do not perform any of the other pre-processing steps.

3.Extract the luminance (Y channel) for the Contrast Test ROI, centred on the centre SFR square, where the dimensions of the ROI should be (0.2 x width of Y channel) by (0.2 x height of Y channel). Label the two data arrays as follows:

Contrast_ROI

For the ROI, perform the following operations:

4.Find the 5th and 95th percentiles of the pixel distributions for the ROI by sorting the entire population. Name these values black_level and white_level.

black_level = value corresponding to the 5th %-ile

white_level = value corresponding to the 95th %-ile

5.Calculate the percentage contrast, contrast, using the following formula:

contrast = (black_level / white_level) * 100

6.Round the final output to two decimal places.

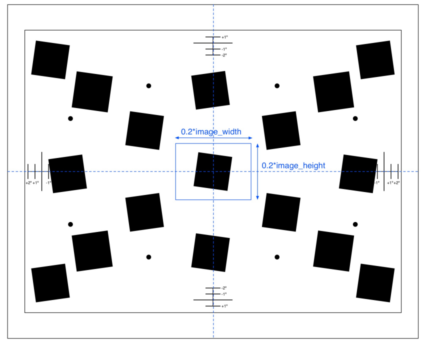


Figure 8: Diagram illustrating the location of the Contrast Test ROI in the High Contrast Transmissive Apple SFR chart

Test Methodology Flow Chart

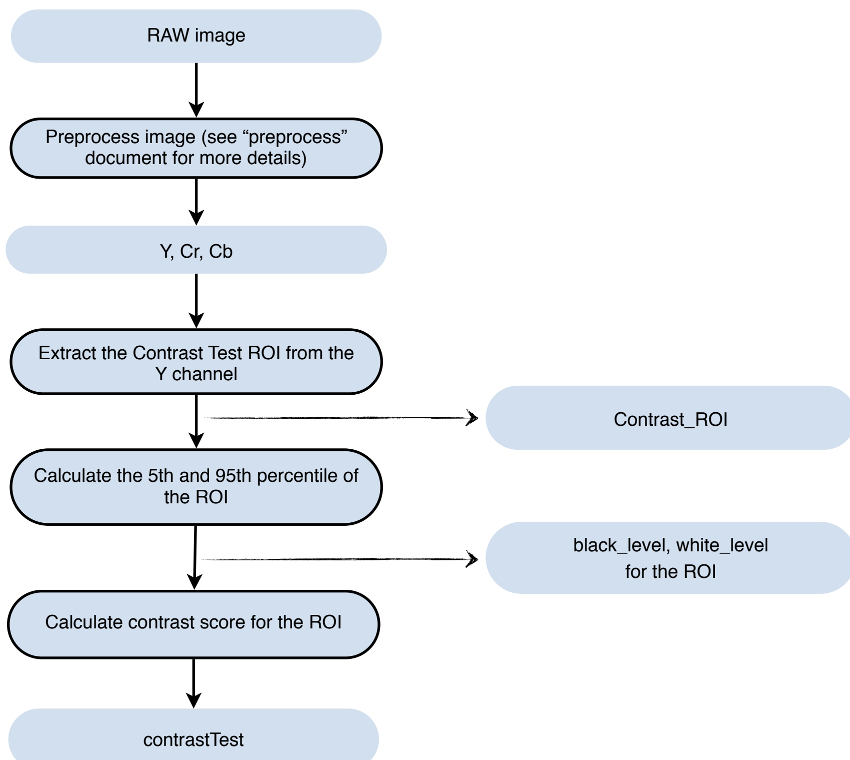


Figure 9 : Contrast Test Methodology Flowchart

Results Naming Convention in Log File

Name	Description	Unit
contrast	Contrast of the Contrast Test ROI	%

Defective Line

Defective line testing is designed to detect any stray rows or columns with unexpected responses. These defects are typically caused by sensor-level defects/damage.

Defective Line Setup

Test Hardware Setup

A D50 light source fitted with an opal diffuser (as per Appendix A) and a dark field should be used as the test targets.

	Light Field	Dark Field
Illuminant	D50	None
Colour temperature	5000K ± 20K	None
Light level	100-300 cd/m	<1 lux
Target	Opal Diffuser	-
Distance to target	8mm - 16mm	-
Orientation	module face up	-

Module Hardware Setup

Parameter	Light Field	Dark Field	Comments
Lens position	focused @ 2m	-	Lens position must be in a deterministic position.
SW Exposure Adjust	80% ±5% FSD	NA	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog Gain	1x	max gain	
Digital Gain	1x	1x	
Integration Time	Variable	66ms	Light field exposure controlled by SW exposure adjust, limited by frame rate
Frame Rate	15fps	15fps	
LSC	Disabled	Disabled	
AWB	Disabled	Disabled	

Parameter	Light Field	Dark Field	Comments
DPC	On		Please refer to the individual ERSs for sensor register settings.

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). Five RAW image captures are to be averaged and saved as a RAW image.

Test Image Specification	Value
Number of image captures	5 RAW frames, which should be averaged
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

1.Capture 5 RAW images, for each of the test conditions outlined above (i.e., light field and dark field).

For each test condition, do the following:

2.Calculate the average of the 5 RAW images.

3.Preprocess the image (see “preprocess” document for more details): subtract pedestal from image, and separate data into the four Bayer channels. Do not perform any of the other pre-processing steps. Bayer channel data should be used for all subsequent calculations.

For each Bayer channel of each of the test images (i.e., light field & dark field), perform the following operations:

4.Calculate the mean value for each row and column. Store each set of values as vectors, with the following names:

colMeans
rowMeans

5.Apply a 1-D averaging filter to the mean vectors using the following kernel: [1/2 0 1/2]. Name the averages as follows:

colMeans_avg
rowMeans_avg

6.Calculate FPN (fixed pattern noise) line metric values as follows, where “/” is used to denote the element-by-element quotient of two vectors:

6.1.light field

$$fpn\_col = 100 \cdot \frac{|colMeans - colMeans\_avg|}{colMeans\_avg}$$

$$fpn\_row = 100 \cdot \frac{|rowMeans - rowMeans\_avg|}{rowMeans\_avg}$$

6.2.dark field

$$fpn\_col = |colMeans - colMeans\_avg|$$

$$fpn\_row = |rowMeans - rowMeans\_avg|$$

7.Discard the first and last values of *fpn_col* and *fpn_row*,, since these values are based on invalid data from the averaging filter function in Step 5.

8.Find and report the maximum value for *fpn_col* and *fpn_row* and the corresponding row (or column) number for this value. (Note: the row and column numbers should correspond to the row & column numbers in the original full-resolution RAW image obtained from the sensor, and not the row & column numbers from the subsampled Bayer channels)

Test Methodology Flow Chart

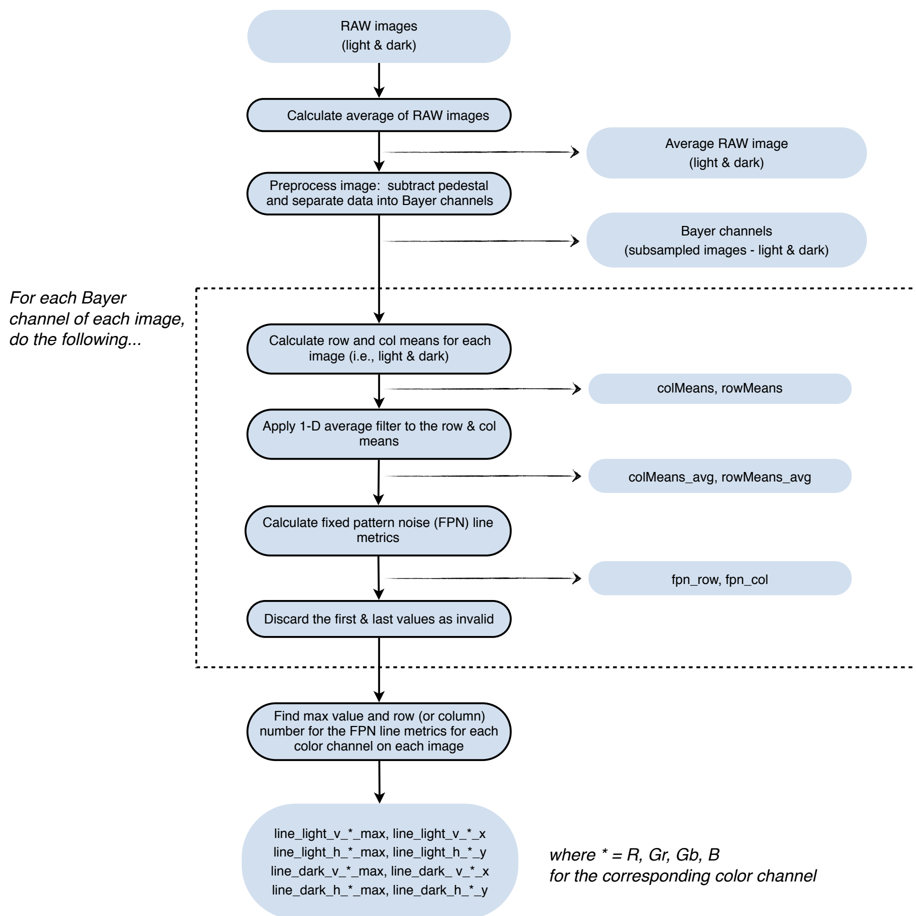


Figure 10 : Defective Line Test Methodology

Results Naming Convention in Log File

Name	Description	Unit
line_light_v_*_max	max *fpn_col* value for the light field image, for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
line_light_v_*_x	column number of the original RAW sensor image corresponding to the value in line_v_light_*_max	-
line_light_h_*_max	max *fpn_row* value for the light field image, for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
line_light_h_*_y	row number of the original RAW sensor image corresponding to the value in line_h_light_*_max	-
line_dark_v_*_max	max *fpn_col* value for the dark field image, for particular colour channel(i.e., replace * by R, Gr, Gb, or B)	LSB
line_dark_v_*_x	column number of the original RAW sensor image corresponding to the value in line_v_dark_*_max	-
line_dark_h_*_max	max *fpn_row* value for the dark field image, for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	LSB
line_dark_h_*_y	row number of the original RAW sensor image corresponding to the value in line_h_dark_*_max	-

Defective Pixels

The defective pixel test is used to catch the pixels which are either stuck at high or low or if they are covered by some material.

Defective Pixel Setup

Test Hardware Setup

A D50 light source fitted with an opal diffuser (as per Appendix A) should be used as the test target for light field testing and dark conditions should be used for dark field testing.

Parameter	Light Field	Dark Field
Illuminant	D50	None
Colour temperature	5000K ± 20K	None
Light level	100-300 cd/m	<1 lux
Target	Opal Diffuser	n/a
Distance to target	8mm -16 mm	n/a
Orientation	Module up	any orientation is acceptable

Module Hardware Setup

Module set up should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Light Field	Dark Field	Comments
Lens position	focused @ 2m	-	Lens position must be in a deterministic position.
SW Exposure Adjust	80% ±5% FSD	NA	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog Gain	1x	max gain	
Digital Gain	1x	1x	
Integration Time	Variable	66ms	Light field exposure controlled by SW exposure adjust, limited by frame rate
Frame Rate	15fps	15fps	
LSC	Disabled	Disabled	

Parameter	Light Field	Dark Field	Comments
AWB	Manual	Disabled	Manual AWB only applied in light field image.
ROI	29x29	29x29	
DPC	as per module ERS		Please refer to the individual ERSs for sensor register settings.

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). Five RAW image captures are to be averaged and saved as a RAW image.

Test Image Specification	Value
Number of image captures	5 RAW frames, which should be averaged into one frame for each condition
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

Defective Pixel Test's input is a full resolution RAW image. To detect defective pixels the analysis is performed on the separate Bayer channels, using the following procedure.

1.Five image captures are to be averaged and saved as a RAW image. Any future reference to RAW image refers to this integer based averaged image. Use the following conditions to get the images.

1.1.Light field test with DPC off (both static and dynamic off)

1.2.Light field test with static on (static DPC on and dynamic DPC off)

1.3.Light field test with DPC on (both static and dynamic on)

1.4.Dark field test with DPC off (both static and dynamic off)

1.5.Dark field test with static on (static DPC on and dynamic DPC off)

1.6.Dark field test with DPC on (both static and dynamic on)

2.Preprocess the image by subtracting the pedestal from image and separate data into four Bayer channels.

2.1.For the light field image, apply AWB as detailed in the preprocess section.

3.Individually smooth the RAW image with a flat, unity Bayer kernel of size 29x29 pixels.

3.1.Edge pixels should be mirrored over the frame boundary to enable the forming of a full 29x29 region of interest for the smoothing filter.

To maintain the Bayer pattern, alternate every other row/column within the padded region. This will allow the 29x29 Bayer filter to be applied to the correct colour planes.

...	...	...	...	...	...	...	...	...	...	...	...
...	22	22	21	21	21	21	22	22	23	23	...
...	22	22	21	21	21	21	22	22	23	23	...
...	12	12	11	11	11	11	12	12	13	13	...
...	12	12	11	11	11	11	12	12	13	13	...
...	12	12	11	11	11	11	12	12	13	13	...
...	12	12	11	11	11	11	12	12	13	13	...
...	22	22	21	21	21	21	22	22	23	23	...
...	22	22	21	21	21	21	22	22	23	23	...
...	32	32	31	31	31	31	32	32	33	33	...
...	32	32	31	31	31	31	32	32	33	33	...
...	...	...	...	...	...	...	...	...	...	...	...

...	...	...	...	...	...	...	...	...	...	...	...
...	22	22	21	21	21	21	22	22	23	23	...
...	22	22	21	21	21	21	22	22	23	23	...
...	12	12	11	11	11	11	12	12	13	13	...
...	12	12	11	11	11	11	12	12	13	13	...
...	12	12	11	11	11	11	12	12	13	13	...
...	12	12	11	11	11	11	12	12	13	13	...
...	22	22	21	21	21	21	22	22	23	23	...
...	22	22	21	21	21	21	22	22	23	23	...
...	32	32	31	31	31	31	32	32	33	33	...
...	32	32	31	31	31	31	32	32	33	33	...
...	...	...	...	...	...	...	...	...	...	...	...

Figure 11 : Example of mirroring and alternating in the top left corner.

3.2. Apply a flat, unity Bayer kernel of size 29x29 on the RAW image.

1	0	1	...	...	...	...	...	...	...	1	0	1
0	0	0	...	...	...	...	...	...	...	0	0	0
1	0	1	...	...	...	...	...	...	...	1	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	1	0	1	0	1	...	...	...	...
...	...	...	...	0	0	0	0	0	...	...	...	...
...	...	...	...	1	0	1	0	1	...	...	...	...
...	...	...	...	0	0	0	0	0	...	...	...	...
...	...	...	...	1	0	1	0	1	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...
1	0	1	...	...	...	...	...	...	...	1	0	1
0	0	0	...	...	...	...	...	...	...	0	0	0
1	0	1	...	...	...	...	...	...	...	1	0	1

Figure 12 : Unity "Bayer" kernel applied to the RAW plane.

3.3. Reduce size of smoothed image back to original resolution.

4.Compare the percentage delta (for the light field) or absolute delta (for the dark field) between the value of each pixel in the data array obtained from step 2 and the smoothed data array obtained from step 3. If the percentage delta for the light field is beyond x% (or the absolute delta is beyond y LSB for the dark field), then the pixel is identified as a defective pixel, where x and y are threshold values specified in the module ERS. Mark all defective pixels as 1 and the remaining pixels as 0.

map_defect is a binary map showing all defective pixels as 1

5.For each defective pixel, create a surrounding 5x5 kernel (as defined in the module ERS) and sum up all the defective pixels within that kernel. If the sum ≥ cluster_size (as defined in the module ERS), then all defective pixels within that window are considered as part of the cluster and should be tagged as a cluster.

map_cluster is a binary map showing all cluster pixels as 1

6.Remove all cluster pixels from the defect_map to avoid double counting defects.

7.Remove all NVM tagged pixels from the defect_map to avoid double correction. Note that any pixel locations within 2 rows/columns from the edge should not be removed from the defect_map.

8.Identify remaining defective pixel detection capability. Note that in the steps below, edge cases should be handled by cropping the ROI as necessary, and adjusting the averaging algorithm to the correct number of pixels

8.1.For each defective pixel P5, create a 5x5 Bayer kernel

P1	**-**	**P2**	**-**	**P3**
-	**-**	**-**	**-**	**-**
P4	**-**	**P5**	**-**	**P6**
-	**-**	**-**	**-**	**-**
P7	**-**	**P8**	**-**	**P9**

Figure 13 : 5x5 Defect Pixel Bayer Kernel

8.2.Identify *tempAvg*

8.2.1.If neighbour_type is “avg” type (as denoted in ERS):
Remove P5 and the highest and lowest pixel values from the neighbouring pixels. Average the remaining six values.

$$\begin{aligned} neighbour \;\; = \;\; & \frac{P1+P2+P3+P4+P6+P7+P8+P9}{6} \\ & - \frac{max(P1,P2,P3,P4,P6,P7,P8,P9)}{6} \\ & - \frac{min(P1,P2,P3,P4,P6,P7,P8,P9)}{6} \end{aligned}$$

8.2.2.If neighbour_type is “delta” type (as denoted in ERS):
Calculate the smallest absolute difference between the P5 and the neighbouring pixels.

$$\begin{aligned} neighbour = \min(&|P1-P5|,|P2-P5|,|P3-P5|, \\ &|P4-P5|,|P6-P5|,|P7-P5|, \\ &|P8-P5|,|P9-P5|) \end{aligned}$$

8.3.Generate a detectability map *mapTemp_detection* with the following labels on the remaining defective pixels:

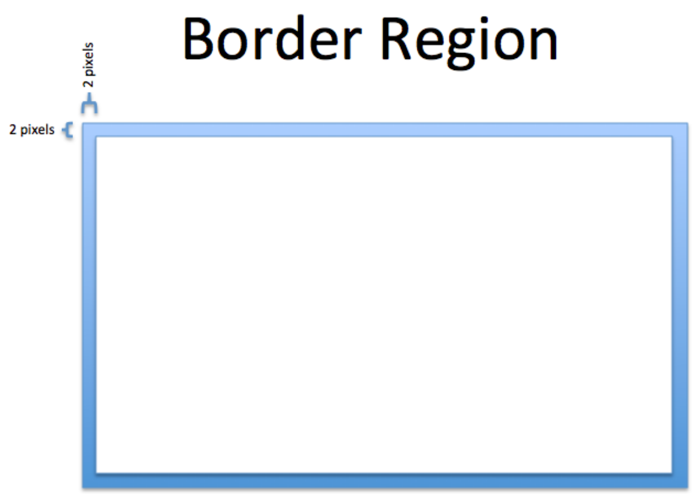
Detectable Pixel (DP) if

P5 > white pixel threshold + neighbour or
P5 < dark pixel threshold - neighbour

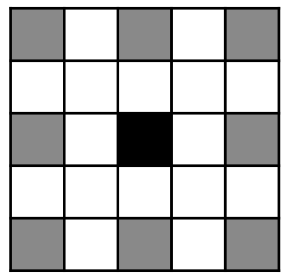
Non-detectable Pixel (NDP)

for all remaining pixels that are not detectable

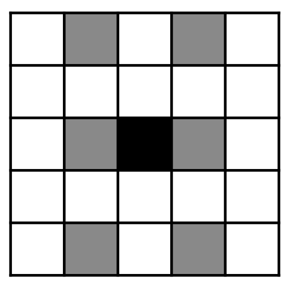
9.Create a binary defect map, *mapTemp_border*, that contains all the DPs and DPPs along the border defined to be two rows and two columns from the edge of the image.



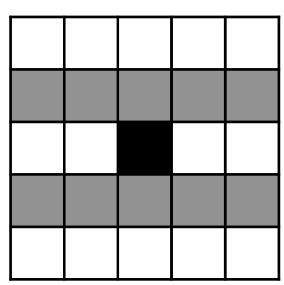
10.Create four new binary defect maps initialized to 0, the same size as the original. Defects get added to the appropriate defect map by applying each of the following two kernels onto each defective pixel (as specified in the ERS): (note that ARPD is only applicable in light field images)



$$\begin{cases} map\_DP & \text{if kernel contains one DP} \\ map\_NDP & \text{if kernel contains one NDP} \\ map\_NDPP & \text{if kernel contains two NDPs} \\ map\_DPP & \text{if kernel contains two defects, and does not satisfy NDPP} \end{cases}$$



$$\begin{cases} map\_NLP & \text{if kernel contains two NDPs} \\ map\_DLP & \text{if kernel contains two defects, and does not satisfty NLP} \end{cases}$$



$$\begin{cases} map\_ARPD & \text{if kernel contains two defects} \end{cases}$$

Figure 14: Defect Kernel Maps

11.(This step is only applicable if a defectivePixel_featureScore_apple function is provided in the program ERS or VSR). Create one new binary defect map initialized to 0, called map_feature, the same size as the original. Defects that were identified as map_DP or map_DPP will be checked against a feature detection algorithm that identifies features based on the raw pixel data from the image. If a pixel is determined to be a feature, it is added to the defect map. If it is not a feature, it remains either a DP or DPP. The algorithm for the feature detection will be provided.

map_feature is a binary map showing all feature pixels as 1

12.(This step is only applicable for light field images.) Compare the percentage delta between the value of each pixel in the image data obtained from step 2 and the smoothed images data obtained from step 3. If the percentage is beyond threshold_defectLow%, the pixel is identified as a defectiveLow pixel. Mark all defectiveLow pixels as 1 and the remaining pixels as 0.

map_defectLow is a binary map showing all defective pixels as 1

13.For each defectiveLow pixel, create a surrounding 5x5 window and sum up all defectiveLow pixels within that window. If the sum ≥ cluster_size, then all defectiveLow pixels within that window are considered as part of the cluster and should be tagged as a cluster.

map_clusterLow is a binary map showing all clusterLow pixels as 1

14. Report the number of defects in each fail map as follows:

DP = number of defects in mapTemp_DP

DPP = number of defects in mapTemp_DPP/2

Border = number of border defects in mapTemp_border

Feature = number of feature defects in mapTemp_Feature

NDP = number of defects in mapTemp_NDP

NDPP = number of defects in mapTemp_NDPP / 2

DLP = number of defects in mapTemp_DLP / 2

NLP = number of defects in mapTemp_NLP / 2

ARPD = number of defects in mapTemp_ARPD /2

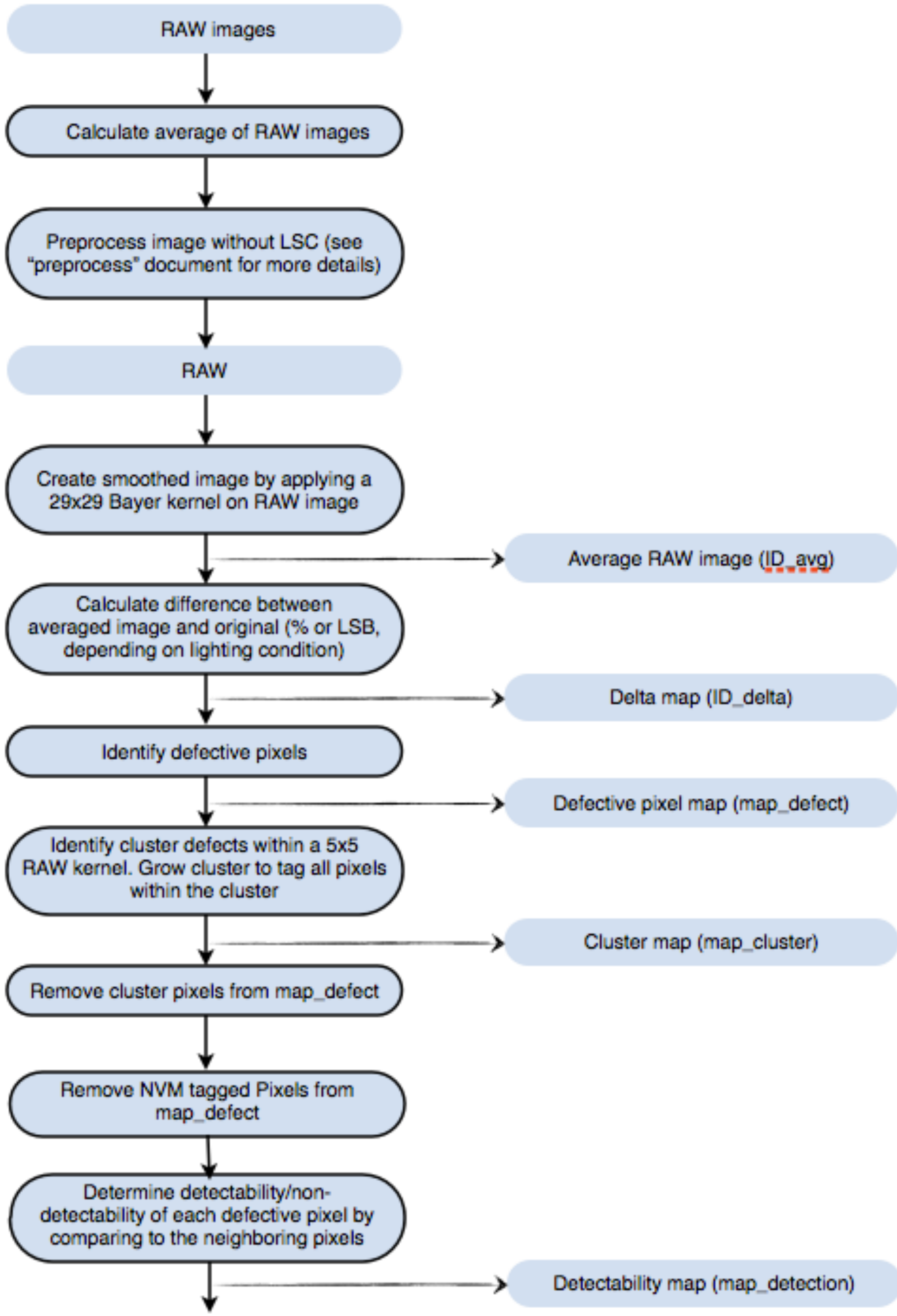
cluster = number of defects in mapTemp_cluster

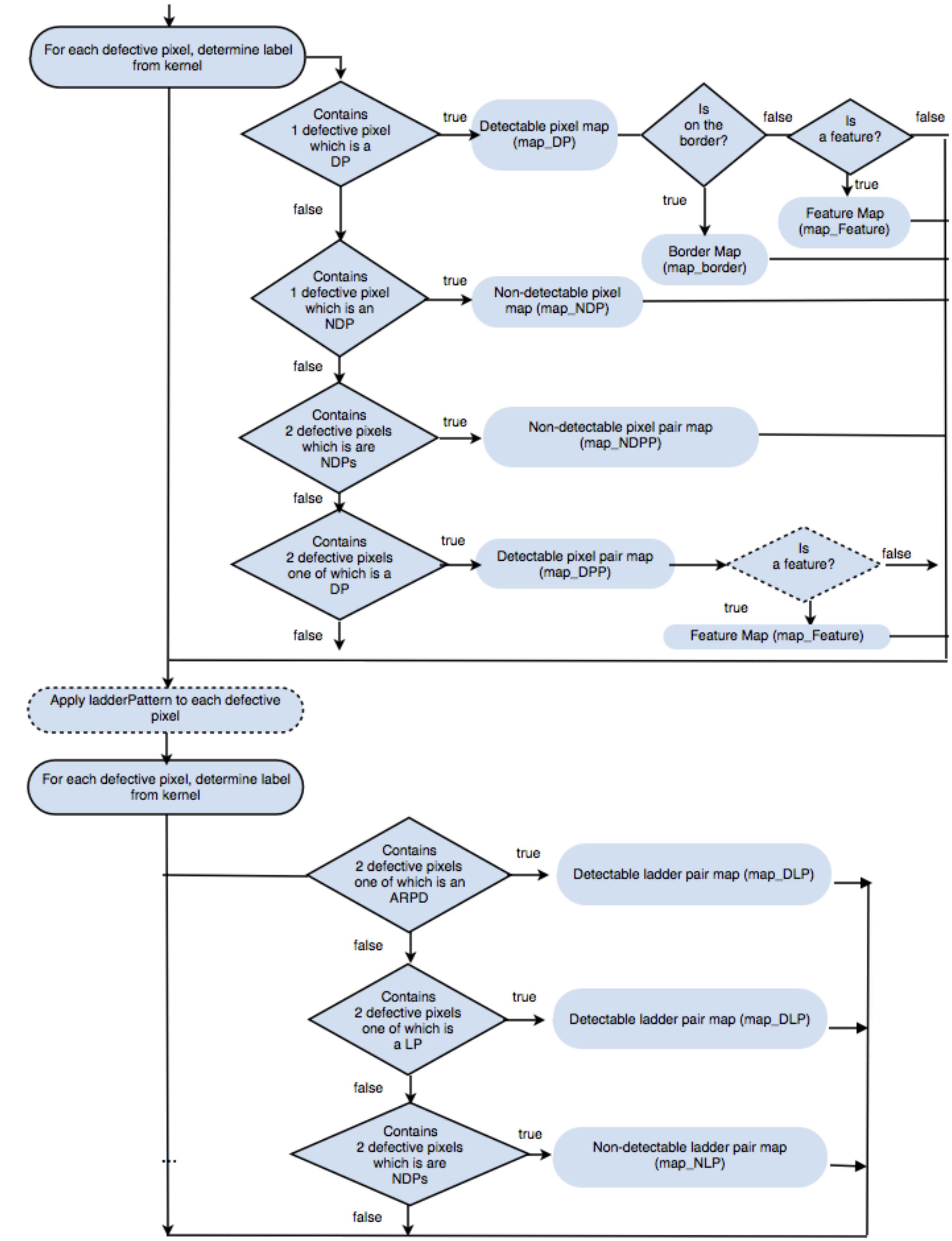
clusterLow = number of defects in mapTemp_clusterLow

15. During the debug process, a final fail map should also be made by combining all previous fail maps into one by applying the following equation per pixel:

```
mapFail(x,y) = max(1*DP(x,y)
                   + 2*DPP(x,y)
                   + 3*border(x,y)
                   + 4*feature(x,y)
                   + 5*NDP(x,y)
                   + 6*NDPP(x,y)
                   + 7*DLP(x,y)
                   + 8*NLP(x,y)
                   + 9*ARPD(x,y)
                   + 10*cluster(x,y)
                   + 11*clusterLow(x,y))
```

Test Methodology Flow Chart




For each defective pixel, determine label from kernel
Contains 1 defective pixel which is a DP
true
Detectable pixel map (map_DP)
Is on the border?
false
Is a feature?
false
true
true
Feature Map (map_Feature)
Border Map (map_border)
false
Contains 1 defective pixel which is an NDP
true
Non-detectable pixel map (map_NDP)
false
Contains 2 defective pixels which is are NDPs
true
Non-detectable pixel pair map (map_NDPP)
false
Contains 2 defective pixels one of which is a DP
true
Detectable pixel pair map (map_DPP)
Is a feature?
false
true
Feature Map (map_Feature)
false
Apply ladderPattern to each defective pixel
For each defective pixel, determine label from kernel
Contains 2 defective pixels one of which is an ARPD
true
Detectable ladder pair map (map_DLP)
false
Contains 2 defective pixels one of which is a LP
true
Detectable ladder pair map (map_DLP)
false
Contains 2 defective pixels which is are NDPs
true
Non-detectable ladder pair map (map_NLP)
false
...

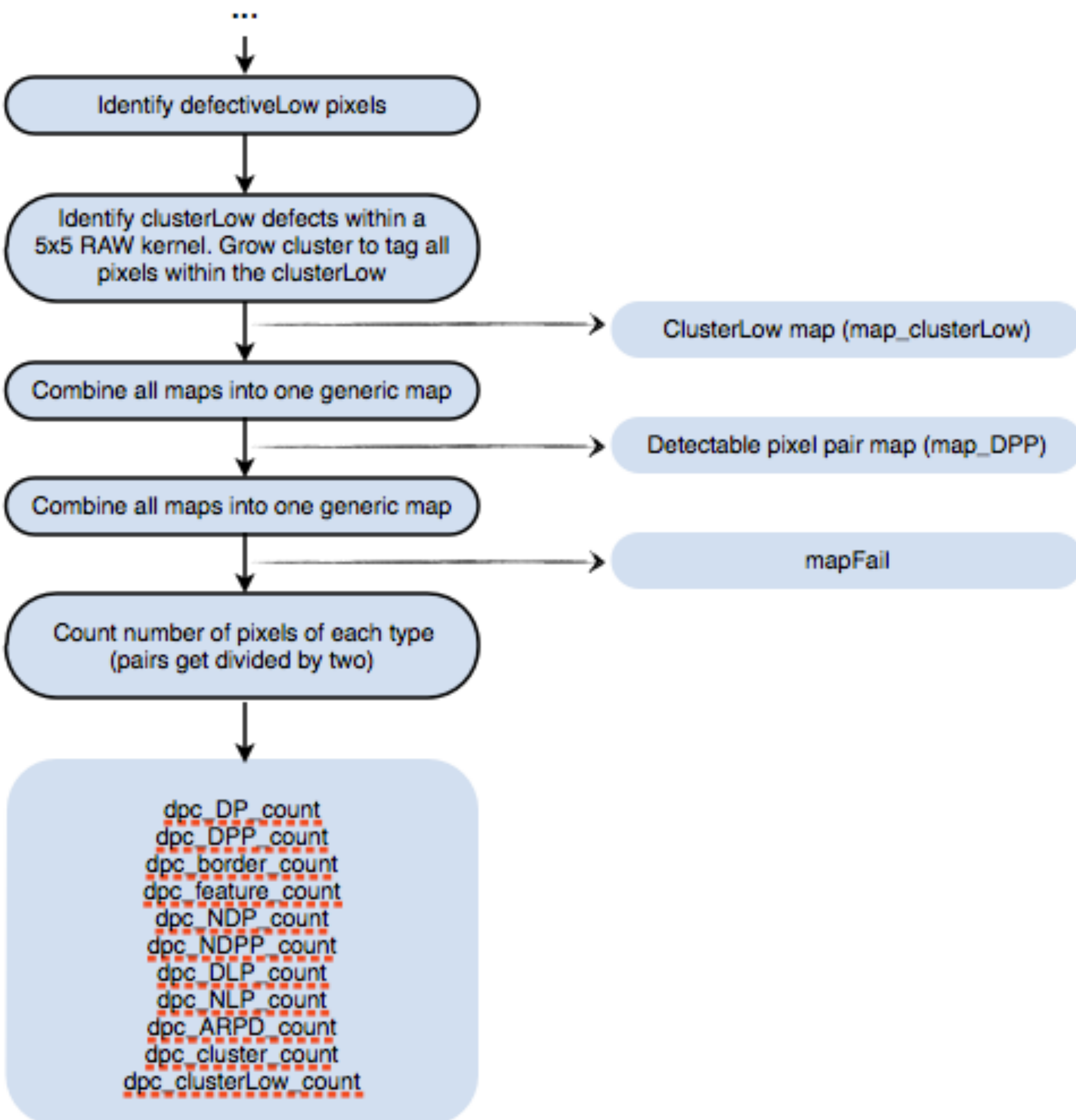


Figure 15 : Defective Pixel Test Methodology Flow Chart

Results Naming Convention in Log File

Note: "*" is to be replaced by the lighting condition per data. (i.e. "dark", "D50" or "A")

Name	Description	Unit
dpc_*_DP_count	count of DP (pixels) from final mapFail	count
dpc_*_DPP_count	count of number of DPP (pairs) from final mapFail	count
dpc_*_border_count	count of pixels in the border region from mapFail (Note: there is no border count for dark lighting condition)	count
dpc_*_feature_count	count of number of features from final mapFail	count
dpc_*_NDP_count	count of NDP (pixels) from final mapFail	count
dpc_*_NDPP_count	count of number of NDPP (pairs) from final mapFail	count
dpc_*_DLP_count	count of DLP (pairs) from final mapFail	count
dpc_*_NLP_count	count of number of NLP (pairs) from final mapFail	count
dpc_*_ARPD_count	count of number of ARPDP (pairs) from final mapFail	count
dpc_*_cluster_count	count of pixels in clusters (pixels) from final mapFail	count
dpc_*_clusterLow_count	count of pixels in clusterLow (pixels) from final mapFail (Note: there is no clusterLow_count for dark lighting condition)	count

Diagonal Field of View

Diagonal field of view (DFOV) measures the worst case diagonal viewing angle in degrees using the Spatial Frequency Response (SFR) test chart imaged by the camera module under test at a given distance.

Diagonal Field of View Setup

Test Hardware Setup

A transmissive Apple SFR chart with D65 light source should be used as the test target.

Parameter	Value
Illuminant	D65
Light Level	>200 cd/m
Target	high contrast transmissive Apple SFR target
Camera distance to target	Specific distances will be set by the camera module ERS. Use the same distances as those specified for measuring SFR. Setup distances must be measured with a calibrated laser distance meter
Orientation	Any orientation is acceptable

Test Chart

The test chart used for measuring DFOV is the same as that used for SFR testing. Please refer to the SFR test and SFR chart documentation for further details.

Module Hardware Setup

Module setup should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Software driven AE	70±5% FSD	average of a 64x64 pixel ROI around the white area which is left of the centre tilted square by 10% of the image width (refer to Figure 1) post pedestal subtraction, on y-channel
Analog gain	1x	
Integration time	<66.67 ms	controlled by AE, limited by frame rate
DPC	off	Refer to specific camera module ERS
Frame rate	15 fps	
LSC	enabled	
AWB	enabled	

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). One RAW image capture is to be saved.

Test Image Specification	Value
Number of image captures	1 image capture.
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

The DFOV calculation is based on a focused SFR image. The four outer field fiducial dots shown in figure 2 provide the basis for calculations. The field dot coordinates in image space have been detected to sub-pixel precision from an image of the SFR test chart. For details on detecting the fiducial dots, please refer to the SFR Test documentation. The final output is rounded to the nearest two decimal places.

Algorithm Notation:

chartDistance_mm : distance from camera to chart
chartSize_mm : total size of the chart area images at nominal for dFOV (mm)
fiducialDistanceOutside_mm : distance between outside fiducials (mm)
actualDoti : fiducial dots in image space (pixel)
actualDot_centre : centre of fiducial dots in image space (pixel)
correctedDoti : modified fiducial dots in image space (mm)
idealCorner: corner point in image space (pixel)
corner_phy : corner point in physical space (mm)
convert_mmPerPixel : millimeter per pixel

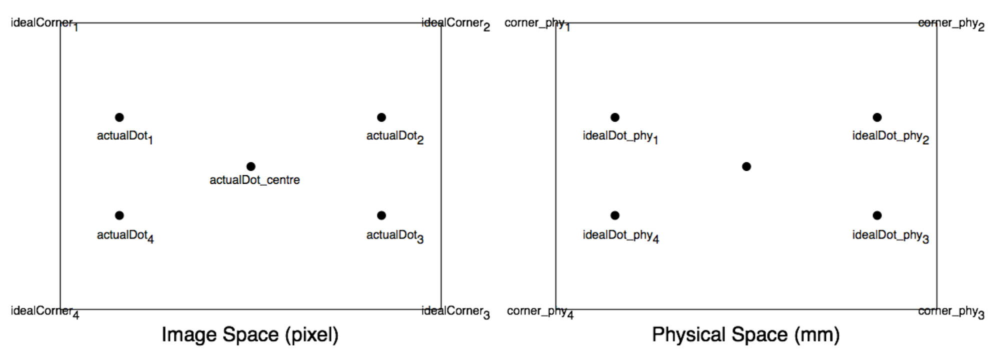


Figure 16 : Four outer field dot fiducials on the Apple SFR chart in the Image Space and Physical Space

Detailed steps of the algorithm are outlined below

1.Preprocess the image (see “preprocess” document for more details): subtract pedestal from image, apply white balance & lens shading correction, apply demosaicing, and convert the image from RGB to YCbCr.

2.Modify the coordinate of field fiducial dots in image space according to rotation

2.1.Calculate the ROT(θ) per ERS

2.2.Modify the field fiducial dots in image space

$$\begin{aligned}
\text{actualDot_centre} &= (\text{actualDot_centre}_x, \text{actualDot_centre}_y) \\
&= \left(\frac{\sum \text{actualDot}_i}{\#\text{ of dots}} \right) \\
&= \left(\frac{\sum \text{actualDot}_{x(i)}}{4}, \frac{\sum \text{actualDot}_{y(i)}}{4} \right)
\end{aligned}$$

$$\text{rotation\_matrix} = \begin{bmatrix} \cos(-\theta) & \sin(-\theta) \\ -\sin(-\theta) & \cos(-\theta) \end{bmatrix}$$

$$\begin{aligned}
\text{correctedDot}_i &= \text{rotation_matrix} \cdot (\text{actualDot}_i - \text{actualDot_centre}) + \text{actualDot_centre} \\
&= \begin{bmatrix} \cos(-\theta) & \sin(-\theta) \\ -\sin(-\theta) & \cos(-\theta) \end{bmatrix} \begin{bmatrix} \text{actualDot}_{x(i)} - \text{actualDot_centre}_x \\ \text{actualDot}_{y(i)} - \text{actualDot_centre}_y \end{bmatrix} + \begin{bmatrix} \text{actualDot_centre}_x \\ \text{actualDot_centre}_y \end{bmatrix}
\end{aligned}$$

3.Calculate convert_mmPerPixel (MP)

calculate values for two diagonals and two horizontals

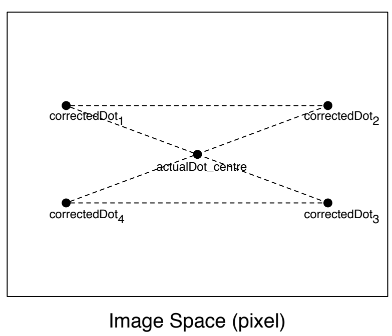


Figure 17 : Lines used to determine conversion factors for mm/pixel

$$\text{convert\_mmPerPixel}_{diagonal} = \left|\frac{\text{physical distance}}{\text{image distance}}\right| = \frac{\sqrt{\text{fiducialDistance\_outside}_x^2 + \text{fiducialDistance\_outside}_y^2}}{\sqrt{(\text{actualDot}_{x(i)} - \text{actualDot}_{x(j)})^2 + (\text{actualDot}_{y(i)} - \text{actualDot}_{y(j)})^2}}$$

$$\text{convert\_mmPerPixel}_{horizontal} = \left|\frac{\text{physical distance}}{\text{image distance}}\right| = \frac{\text{fiducialDistance\_outside}_x}{\sqrt{(\text{actualDot}_{x(i)} - \text{actualDot}_{x(j)})^2 + (\text{actualDot}_{y(i)} - \text{actualDot}_{y(j)})^2}}$$

Note the final value is taken as an average from the four lines.

Calculate the corner point in physical space

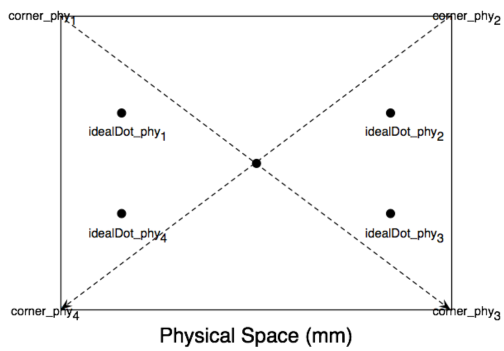


Figure 18 : Lines used to calculate physical corners

$$\text{idealDot\_phy}_i = \left(\frac{\text{chartSize\_mm}_x}{2} \pm \frac{\text{fiducialDistanceOutside\_mm}_x}{2}, \frac{\text{chartSize\_mm}_y}{2} \pm \frac{\text{fiducialDistanceOutside\_mm}_y}{2}\right)$$

$$\text{corner\_phy}_i = \text{idealDot\_phy}_i + \text{convert\_mmPerPixel}(\text{idealCorner}_i - \text{correctedDot}_i)$$

4.Calculate the diagonal field of view (dFOV)

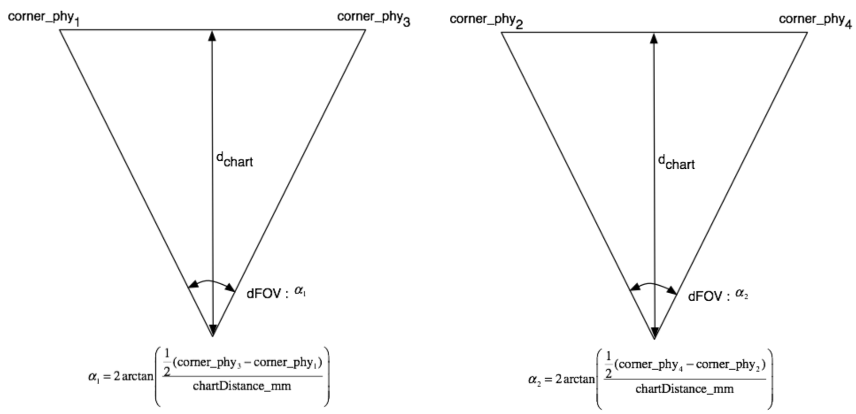


Figure 19 : α angles calculated from physical corners

5.report the minimum for dFOV

$$\alpha = \min(\alpha 1, \alpha 2)$$

Test Methodology Flow Chart

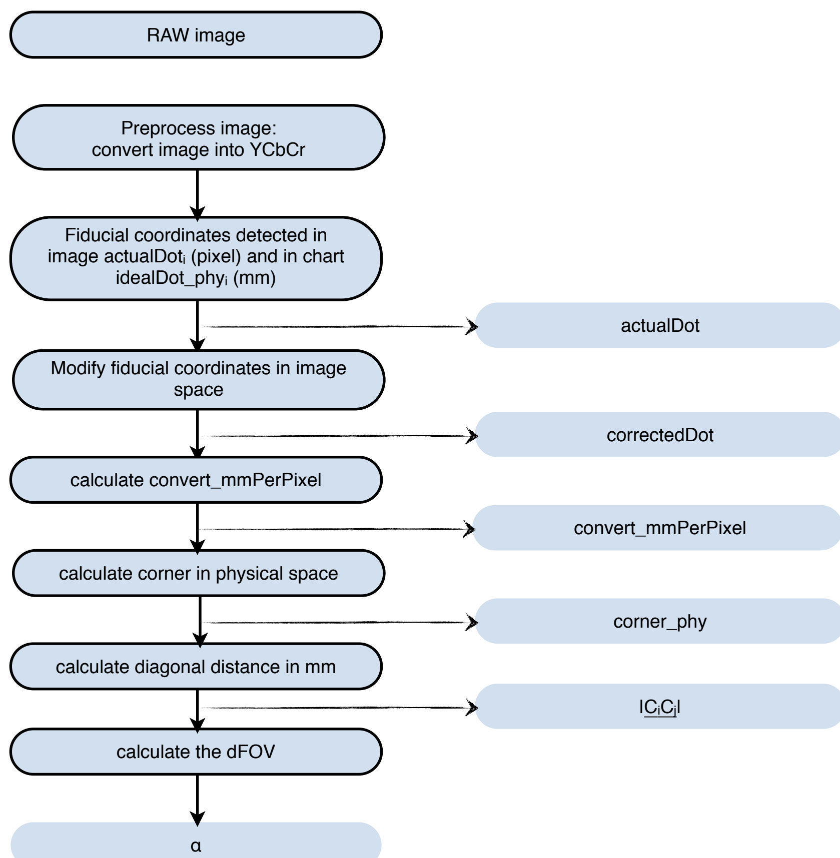


Figure 20 : Diagonal Field Of View Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
alignment_dFOV	Diagonal field of view	°

Dark Signal Non-Uniformity

This document defines the Dark Signal Non-Uniformity (DSNU) test setup and test methodology to be used for screening dark non-uniformity caused by abnormal shading behaviour from dark current and residual sensor surface charge. The dark images shall be captured and processed to calculate the dark signal non-uniformity metrics described in Test Algorithm section.

Dark Signal Non-Uniformity Setup

Test Hardware Setup

Parameter	Value
Illuminant	None (dark condition)
Correlated colour temperature	-
Light level	<1 lux
Target	None (dark condition)
Camera distance to target	-
Orientation	any orientation is acceptable

Module Hardware Setup

Module set up should follow module ERS default settings. In addition, the settings below supersede any default settings:

Note that a ~2.5s warmup must precede before this image capture.

	value	comments
software driven AE	None	
analog gain	max gain	
integration time	233 ms	
DPC	on	Refer to specific camera module ERS
frame rate	4 fps	
LSC	disabled	
AWB	disabled	

Test Image

The test images should be RAW images captured at the maximum bit-depth (e.g., 10-bit). Five RAW image captures are to be averaged and saved as a RAW image.

Test Image Specification	Value
Number of image captures	5 RAW frames, which should be averaged into one frame for each condition
Image bit-depth	Full-scale bit depth (e.g., 10-bit)
Final image format for algorithm calculations	RAW Bayer (R, Gr, Gb, B) image

Test Methodology

Preface: Final output should be rounded to two decimal places. Below indexing conventions follow (i,j) when referring to pixel and (m,n) when referring to ROI.

1. Average five RAW images, captured under the test conditions outlined above.

2. Preprocess the image (see “preprocess” document for more details): subtract pedestal from image, do not apply lens shading correction or white balance. Separate the image into Bayer Channels, R, Gb, Gr and B.

3. Combine Gr and Gb colour planes into one G colour plane.

$$G(i,j) = \frac{\left(Gr(i,j) - Gb(i,j)\right)}{2}$$

R, G, and B channel data should be used for all subsequent calculations.

4. Decimate the image for each bayer channel. Each bayer channel is divided into multiple ROIs, where the appropriate ROI size is defined by the module ERS (example sizes listed in Algorithm Parameters Setting Table below). In the case the image size is not divisible by the ROI size, the residual rows and columns will be evenly assigned to the border ROIs (if odd in number, the top and/or left ROIs will have the extra row/column respectively), resulting in the centre ROIs having the size as specified by ROI size. The border ROIs may have a few more rows and/or columns. After dividing the image into multiple ROIs, the pixel values are averaged for each ROI per colour channel to obtain R, G, and B.

5. Calculate the range parameters, defined by the difference between the maximum and minimum ROI for each bayer channel.

$$dsnu\_range\_R = \max(R) - \min(R)$$
$$dsnu\_range\_G = \max(G) - \min(G)$$
$$dsnu\_range\_B = \max(B) - \min(B)$$

6. Calculate the signal for each ROI, defined as the Euclidean distance of R, G, B bayer channels.

$$signal(m,n) = \sqrt{R(m,n)^2 + G(m,n)^2 + B(m,n)^2}$$

7. Report the maximum signal value.

$$dsnu\_signalMax = \max\left(signal(m,n)\right)$$

8. For each ROI per bayer channel, calculate the delta between the ROI and the ROI at the maximum signal location.

$$deltaSignal(m,n) = \sqrt{\begin{array}{l}\left(R(m,n) - R(signalMax)\right)^2 + \\ \left(G(m,n) - G(signalMax)\right)^2 + \\ \left(B(m,n) - B(signalMax)\right)^2\end{array}}$$

$$dsnu\_deltaSignalMax = \max\left(deltaSignal(m,n)\right)$$

9. Report the maximum delta signal value.

10. Based on *dsnu_deltaSignal_max*, a tolerance criteria for region growing signal is selected from *growTolerance*, where *growTolerance* (1-by-3 matrix) and *lowDeltaSignal* are adjustable parameters defined by Apple, based on control run data.

11. Perform region growing on *signal* using coordinates of *dsnu_signal_max* and tolerance criteria selected in step 10 to obtain *signalMap*, *dsnu_sumSignal*, and *signalArea*. For this test, region growing grows one region starting from seed ROI value *dsnu_signal_max*. The region is iteratively grown by comparing all allocated neighbouring ROI values as shown below to the region mean using similarity metric to determine whether the neighbour should be added to the region. The process continues until similarity exceeds tolerance criteria.

Neighbor 1 (m-1, n-1)	Neighbor 2 (m-1, n)	Neighbor 3 (m-1, n+1)
Neighbor 4 (m, n-1)	ROI_toConsiderGrowing (m, n)	Neighbor 5 (m, n+1)
Neighbor 6 (m+1, n-1)	Neighbor 7 (m+1, n)	Neighbor 8 (m+1, n+1)

$$similarity = \frac{abs\left(signal(Neighbor\_n) - mean(region)\right)}{mean(region)}$$

$$signalMap(m,n) = \begin{cases} 1 & if\ signal(m,n)\ part\ of\ grown\ region, \\ 0 & else. \end{cases}$$

$$signalArea = \sum_{m\_height,\, n\_width} signalMap(m,n)$$

12. Assign *dsnu_deltaSignal_max* to *dsnu_sumSignal* if *dsnu_deltaSignal_max* does not exceed two times *lowDeltaSignal* an *signalArea* is 1 or *signalArea* divided by *signal* size (*m_height*n_width*) is larger than *extent*, where *extent* is an adjustable parameter defined by Apple, based on control run data.

13. Report *dsnu_sumSignal* value.

14. For each ROI, calculate the local bayer channel difference between the ROI under test and each of its surrounding neighbours as shown below. Find the maximum local bayer channel difference value for each ROI.

Neighbour 1 (m-1, n-1)	Neighbour 2 (m-1, n)	Neighbour 3 (m-1, n+1)
Neighbour 4 (m, n-1)	ROI_underTest (m, n)	Neighbour 5 (m, n+1)
Neighbour 6 (m+1, n-1)	Neighbour 7 (m+1, n)	Neighbour 8 (m+1, n+1)

$$deltaSignalLocal(m,n) = \max\left(\sqrt{\begin{array}{l}(R(m,n) - R(neighbour)^2 + \\ (G(m,n) - G(neighbour)^2 + \\ (B(m,n) - B(neighbour)^2\end{array}}\right)$$

where neighbour represents each of Neighbour 1... 8

Border ROIs may consider less than 8 neighbours. As an example, for the most top right ROI, only 3 neighbours need to be considered as shown below.

Neighbor 1 (1, w-1)	ROI_underTest (1,w)
Neighbor 2 (2, w-1)	Neighbor_3 (2,w)

15. Separate deltaSignalLocal into three regions centre, corner, and edge using predefined corner size specified in the module ERS (example sizes listed in Algorithm Parameters Setting Table below).

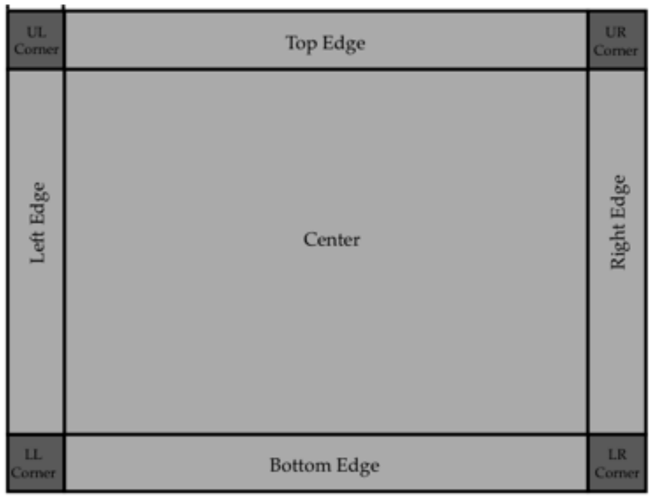


Figure 21 : Partition of centre, corner, and edge of an image

16.Calculate the following three metrics.

dsnu_deltaSignalLocal_cen_max - maximum deltaSignalLocal of the centre region
dsnu_deltaSignalLocal_edg_max - maximum deltaSignalLocal of the edge region
dsnu_deltaSignalLocal_corner_max - maximum deltaSignalLocal of the corner region

17.Record the (x,y) coordinates for the following parameters.

dsnu_range_R
dsnu_range_G
dsnu_range_B
dsnu_signal_max
dsnu_deltaSignal_max
dsnu_deltaSignalLocal_cen_max
dsnu_deltaSignalLocal_edge_max
dsnu_deltaSignalLocal_corner_max

Where “x” and “y” are the coordinates of the centre pixel of each ROI, mapped back to the coordinate system of the full resolution RAW image.

ex) [m n] = a location in down sampled image in 1 bayer channel
consider 3 bayer channels to be 3 separate images, R, G, B
x location in RAW image = 2*(ROIsizeX*m - round(ROIsizeX/2))
y location in RAW image = 2*(ROIsizeY*n - round(ROIsizeY/2))
ROIsizeX is the ROI size in the x direction.
ROIsizeY is the ROI size in the y direction.

Test Methodology Flow Chart

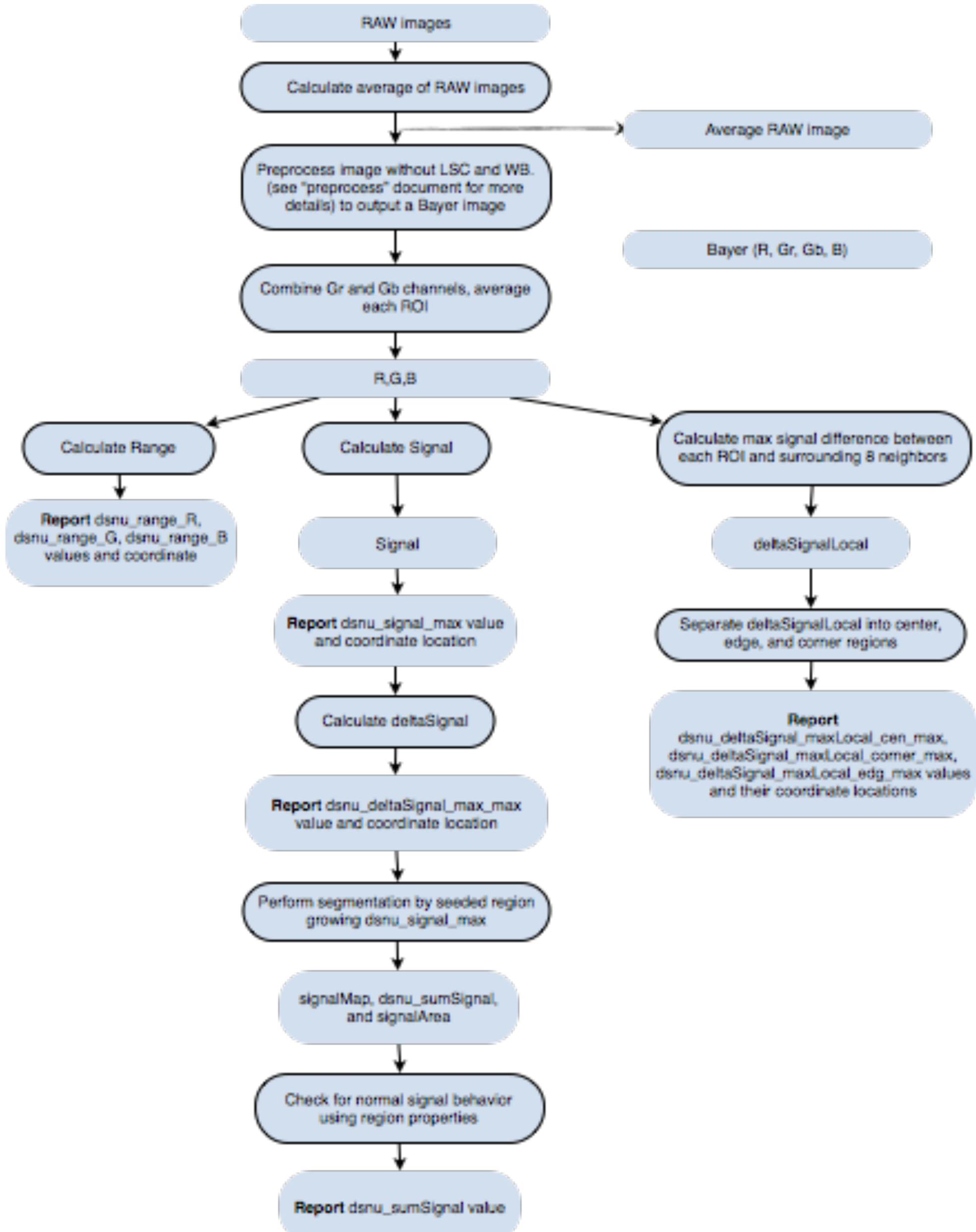


Figure 22 : DSNU Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
dsnu_range_R	Difference between max and min value of R channel	LSB
dsnu_range_G	Difference between max and min value of G channel	LSB
dsnu_range_B	Difference between max and min value of B channel	LSB
dsnu_signal_max	Max signal value	LSB
dsnu_deltaSignal_max	Max delta signal value	LSB
dsnu_sumSignal	Sum of segmented region signal values	LSB
dsnu_deltaSignalLocal_cen_max	Max deltaSignalLocal value in centre region	LSB
dsnu_deltaSignalLocal_edg_max	Max deltaSignalLocal value in edge region	LSB
dsnu_deltaSignalLocal_corner_max	Max deltaSignalLocal value in corner region	LSB
dsnu_range_R_max_x	x coordinate (mapped back to full-resolution image) of max R channel value	pixel
dsnu_range_R_max_y	y coordinate (mapped back to full-resolution image) of max R channel value	pixel
dsnu_range_G_max_x	x coordinate (mapped back to full-resolution image) of max G channel value	pixel
dsnu_range_G_max_y	y coordinate (mapped back to full-resolution image) of max G channel value	pixel
dsnu_range_B_max_x	x coordinate (mapped back to full-resolution image) of max B channel value	pixel
dsnu_range_B_max_y	y coordinate (mapped back to full-resolution image) of max B channel value	pixel
dsnu_range_R_min_x	x coordinate (mapped back to full-resolution image) of min R channel value	pixel
dsnu_range_R_min_y	y coordinate (mapped back to full-resolution image) of min R channel value	pixel
dsnu_range_G_min_x	x coordinate (mapped back to full-resolution image) of min G channel value	pixel
dsnu_range_G_min_y	y coordinate (mapped back to full-resolution image) of min G channel value	pixel
dsnu_range_B_min_x	x coordinate (mapped back to full-resolution image) of min B channel value	pixel

Name	Description	Unit
dsnu_range_B_min_y	y coordinate (mapped back to full-resolution image) of min B channel value	pixel
dsnu_signal_max_x	x coordinate (mapped back to full-resolution image) of maxSignal value	pixel
dsnu_signal_max_y	y coordinate (mapped back to full-resolution image) of maxSignal value	pixel
dsnu_deltaSignal_max_x	x coordinate (mapped back to full-resolution image) of deltaSignal value	pixel
dsnu_deltaSignal_max_y	y coordinate (mapped back to full-resolution image) of deltaSignal value	pixel
dsnu_deltaSignalLocal_cen_max_x	x coordinate (mapped back to full-resolution image) of deltaSignalLocal in centre region	pixel
dsnu_deltaSignalLocal_cen_max_y	y coordinate (mapped back to full-resolution image) of deltaSignalLocal in centre region	pixel
dsnu_deltaSignalLocal_edg_max_x	x coordinate (mapped back to full-resolution image) of deltaSignalLocal in edge region	pixel
dsnu_deltaSignalLocal_edg_max_y	y coordinate (mapped back to full-resolution image) of deltaSignalLocal in edge region	pixel
dsnu_deltaSignalLocal_corner_max_x	x coordinate (mapped back to full-resolution image) of deltaSignalLocal in corner region	pixel
dsnu_deltaSignalLocal_corner_max_y	y coordinate (mapped back to full-resolution image) of deltaSignalLocal in corner region	pixel

Electrical Validation

The electrical validation test procedure goes through a series of basic electrical functionality tests to ensure that the module has not received or undergone gross damage that could put the integrity of the device at risk. The specifications and limits for these tests are defined in the ERS and determine key operating metrics of the device.

Electrical Validation Setup

Test Hardware Setup

Function	Equipments
Power Supply - Voltage	Voltage Measurements
Open/Short - Diode Test	Power Supply, Voltage measurement or Diode check
I^2C	Camera Module, I

Module Hardware Setup

All tests should be performed under the following two test conditions:

1. Still image mode, streaming in preview at the maximum frame rate available (e.g., 15 FPS).
2. Video streaming mode, streaming at the maximum frame rate available (e.g., 24 FPS).

Power Consumption: Current Test

Current consumption of the camera module should be measured on the following rails:

Rails	Modes to test
Analog	Operating, Standby
Digital	Operating, Standby
AFvdd	Operating, Standby
Vddio	Operating, Standby

Operating Mode:

Operating current is defined as when the camera is streaming video at full resolution.

Standby Mode:

Standby current is defined as when the camera is powered on, but is in a hard standby mode with no streaming. Entry into this mode is defined by the ERS.

These measurements need to be taken with, for instance, a multimeter or a current probe (e.g. Agilent N2782B). With either method, the measurement resolution should be 10 µA.

These measurements need to be taken with, for instance, a multimeter or a current probe (e.g. Agilent N2782B). With either method, the measurement resolution should be 10 µA.

An alternative to using the current loop is to put a current measuring device in the circuitry (as shown in the following figure); however, caution must be taken as this method is more invasive and might generate a voltage drop in the power rail.

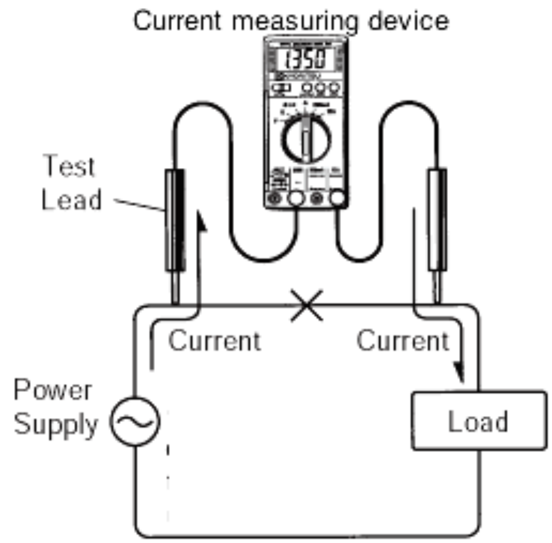


Figure 23: Current Measurement Schematic

Short/Open - Diode Test (Input Pads)

ESD diodes can be tested by sourcing a small current (~1mA) from ground (GND) to the input pads and then using the same terminals to measure a voltage drop across the GND and the input pads. The voltage should conform to the value in the table below:

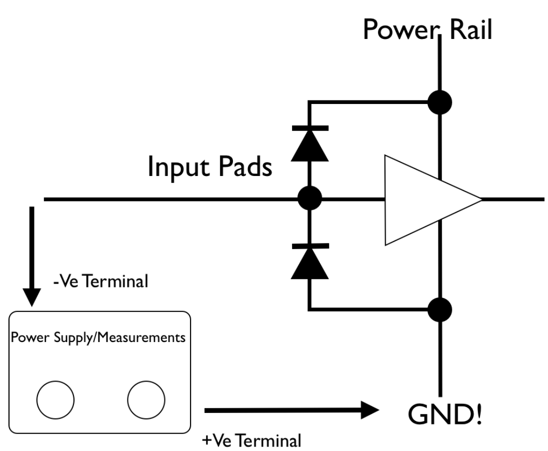


Figure 24 : Typical ESD Configuration on an Input Pad

The current source should follow the following specifications:

Source	Specs
Current type	DC
Minimum accuracy	10uA
Current Rating	700uA (+/-10uA)
Voltage Measurement	1mV

This test needs to be implemented for the following test pairs:

GND/Power	Inputs Signals
AF_GND	AF
DGND	I2C,SHUTDOWN,CAM_CLK,STROBE, and MIPI
DGND	DVDD

It is important to note that AF_GND and GND are not necessarily the same. Therefore, all AF (auto-focus) related signals need to be tested against AF_GND; and all other signals, including digital components, need to be tested against GND.

Measurement Item	Fail
Resistance	< 0.4 (low voltage drop)
Resistance	∞ (0 current draw)
Large Voltage/Resistance deviation	T.B.D.

Name	Description	Unit
ee_opr_analog_current	analog operating current	mA
ee_opr_digital_current	digital operating current	mA
ee_opr_io_current	IO operating current	mA
ee_opr_af_current	AF operating current	mA
ee_stby_analog_current	analog standby current	uA
ee_stby_digital_current	digital standby current	uA

Name	Description	Unit
ee_stby_io_current	IO standby current	uA
ee_stby_af_current	AF standby current	uA
ee_voltage_xxx	diode test will have a trend per pad number (denoted by xxx)	V
ee_pad_fwdResistance_xxx	diode test will have a trend per pad number (denoted by xxx)	Ohms
ee_pad_revResistance_xxx	diode test will have a trend per pad number (denoted by xxx)	Ohms

I2C Test

The I2C bus can be indirectly evaluated by toggling modes of operation and changing the frame rate. After each operation, registers bits should be read out using the I2C interface and compared against the data sheet for validation.

MIPI Data Integrity Test

The data transfer and data integrity of the chip can be tested by using the CRC capability of a MIPI compliant receiver and the chip. If the CRC capability is not available, similar coverage can be obtained by comparing each pixel of an image against a reference test pattern image taken from each module.

The module should be configured to take the CRC from an image capture taken during the first SFR test on the first test machine sequence. This test requires only a single image capture with no frame averaging and standard module conditions.

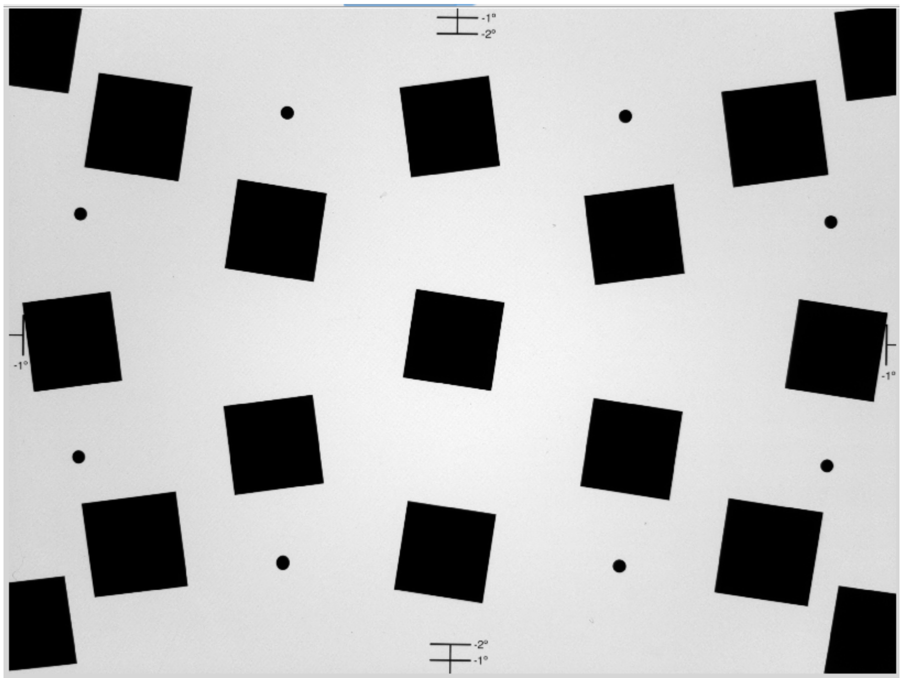


Figure 25: Sensor SFR Test Pattern Image

To use the CRC functionality, the final MIPI line packet should have the CRC checked against the receiver's received CRC for a single frame captured using the SFR image capture. The MIPI CRC calculation is defined in Version 1.00 29-Nov-2005 MIPI Alliance Standard for CSI-2 on page 50 as the following reference C code shown in Figure 26:

```
#define POLY 0x8408   /* 1021H bit reversed */

unsigned short crc16(char *data_p, unsigned short length)
{
      unsigned char i;
      unsigned int data;
      unsigned int crc = 0xffff;

      if (length == 0)
            return (unsigned short)(crc);
      do
      {
            for (i=0, data=(unsigned int)0xff & *data_p++;
              i < 8;i++, data >>= 1)
            {
                  if ((crc & 0x0001) ^ (data & 0x0001))
                        crc = (crc >> 1) ^ POLY;
                  else
                        crc >>= 1;
            }
      } while (--length);

      // Uncomment to change from little to big Endian
//    crc = ((crc & 0xff) << 8) | ((crc & 0xff00) >> 8);

      return (unsigned short)(crc);
}
```

Figure 26 : CRC Calculation Reference Code

The CRC extracted from the transmission is defined on page 49 of the same spec, shown in Figure 27. If MIPI non-continuous mode is used then the CRC is taken from each line as the last two packets in the line. If MIPI continuous mode is used then the CRC is taken as the last two packets from the frame transmission.

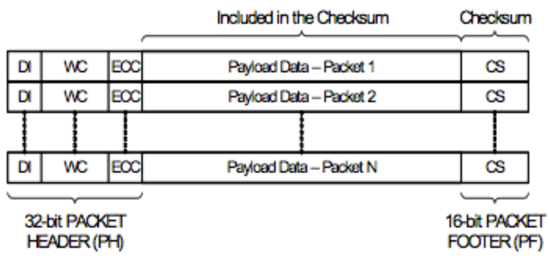


Figure 27 : MIPI CRC Packet Extraction

To use the per-pixel comparison, the decoded RAW bayer image should have every pixel value checked against a known reference colour bar test pattern image specified in the ERS.

If using the CRC, the pattern is considered passing when the decoded CRC matches the calculated CRC. If using per-pixel comparisons, the pattern is considered passing when the image has no pixels deviating from the reference image.

Results Naming Convention in Log File

Name	Description	Unit
ee_testPattern	0 if fail, 1 if pass	

Fixed Pattern Noise

Fixed pattern noise refers to non-temporal spatial noise that can be visualized as a vertical or horizontal strip, band or block.

Fixed Pattern Noise Setup

Test Hardware Setup

Parameter	Value
Illuminant	None (dark condition)
Correlated colour temperature	-
Light level	<1 lux
Target	None (dark condition)
Camera distance to target	-
Orientation	any orientation is acceptable

Module Hardware Setup

Module setup should follow module ERS default settings. In addition, the settings below supersede any default settings:

	Value	Comments
software driven AE	Off	
analog gain	max gain	Maximum sensor gain (e.g., 16x) should be used
integration time	66ms	
DPC	on	Turn both dynamic and static DPC on
frame rate	15 fps	
LSC	disabled	
AWB	disabled	

Test Images

The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should be used to reduce temporal noise.

Test Image Specification	Value
Image captures	5 frames average
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

FPN should be measured under dark conditions. Final output may be rounded to three decimal places.

1. Average 5 RAW images, captured under the test conditions listed above.
2. Preprocess the image (see “preprocess” document for more details): subtract pedestal from the image, but do not perform any of the other pre-processing steps.
3. Calculate the mean of the whole image (on RAW image).
4. Calculate the standard deviation of the image (on RAW image).
5. Calculate the average row and column values per bayer channel.
6. Calculate the standard deviation of the row and column means per bayer channel.

Test Methodology Flow Chart

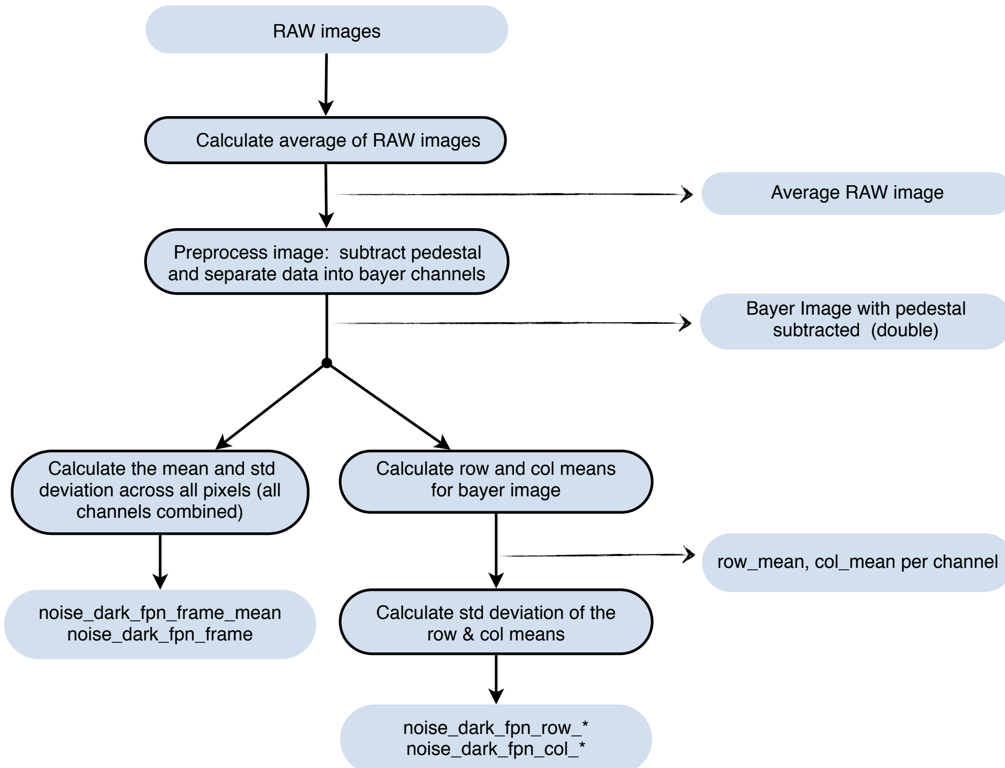


Figure 28 : FPN Test Methodology flow chart

Results Naming Convention in Log File

Name	Description	Unit
noise_dark_fpn_frame_mean	mean of the whole RAW image	LSB
noise_dark_fpn_frame	standard deviation of the whole RAW image	LSB
noise_dark_fpn_row_*	standard deviation of the row averages	LSB
noise_dark_fpn_col_*	standard deviation of the column averages	LSB

Fixed Pattern Noise-Wide

This document defines the Column/Row Noise test setup and test methodology to be used for screening relatively wide column/row noise. The flat field images shall be captured and processed by the FPNwide detection algorithm described in Test Algorithm section to calculate the FPNwide scores.

FPN Wide Setup

Test Hardware Setup

A D50 light source fitted with an opal diffuser (as per Appendix A) should be used as the test target.

Parameter	Value
Illuminant	D50
Correlated colour temperature	5000K
Light level	100 - 300 cd/m
Target	diffuser
Camera distance to target	8 - 16mm
Orientation	module up

Module Hardware Setup

Module set up should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Lens position	focused @ 2m	Lens position must be in a deterministic position.
Software driven AE	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog gain	1x	
Integration time	<66.67 ms	Controlled by SW exposure adjust to meet LSB target, limited by frame rate
DPC	on	Refer to specific camera module ERS
Frame rate	15 fps	
LSC	disabled	
AWB	disabled	

Test Image

The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should be used to reduce temporal noise.

Test Image Specification	Value
Image captures	5 frames average
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

Below are the image processing steps to derive the FPNwide scores (see algorithm flowchart and implementation details in the MATLAB code provided in a separate package). All calculations are to be done with double-precision math. Final output should be rounded to two decimal places.

For each Bayer channel, perform the following operations:

1. Divide the Bayer channel into multiple ROIs, where the appropriate ROI size is defined by the module ERS (e.g., 9-by-9 Bayer pixels), and the pixels in each ROI are averaged. In the case where the array size is not evenly divisible by the size of the ROI, the residual rows and columns should be evenly assigned to the border ROIs, resulting in border ROIs that may have a few more rows and/or columns than the other ROIs.
2. Calculate the average value for each set of row and column ROIs. Store each set of values as vectors with the following names:

   superRow
   superCol

3. In preparation for spatial filtering in Step 4, pad the borders of *superRow* and *superCol* by the appropriate number of pixels (e.g., 2 pixels for a 5-element filter) via linear extrapolation using the first (or last) pair of data points.
4. Perform 1-D spatial convolution on the padded *superRow* and *superCol* curves, respectively, with a specially designed 1-D filter kernel (h=[1/2, 0, ..., 0, -1, 0, ..., 0, 1/2]), where the appropriate filter length is defined by the module ERS (e.g., 5 elements would correspond to h = [1/2, 0, -1, 0, 1/2]).
5. Crop the filtered *superRow* and *superCol* vectors back to the same size they were before padding.
6. Calculate a smoothed version of the curve by applying a 1-D median filter, using the same filter length as Step 4 (e.g., 1x5), on the filtered results from Step 5.
7. Perform an element-by-element division of the filtered results from Step 5 by the smoothed results from Step 6, then multiply the results by a factor of 100.
8. Compute the local deviation curve by calculating (for each element) the maximum of the absolute difference between that element and its n closest neighbors along the curve, where n = filter length + 1 (e.g., for a 5-element filter, the closest 6 neighbors would be the 3 to the left and the 3 to the right of the element).

9. Separate the edge region from the centre region using an appropriate edge size, defined by the module ERS (for example, 3 ROIs).

10. Calculate edge and centre metrics (see Results Naming Convention section below for the complete list of metrics) based on the results from Step 9.

11. Report the row (or column) number for the FPNwide line metrics, as described in the Results Naming Convention section. The row (or column) number should correspond to the subsampled Bayer image. (*Note: In the event of multiple locations of the same max value, only report the first.)

Test Methodology Flow Chart

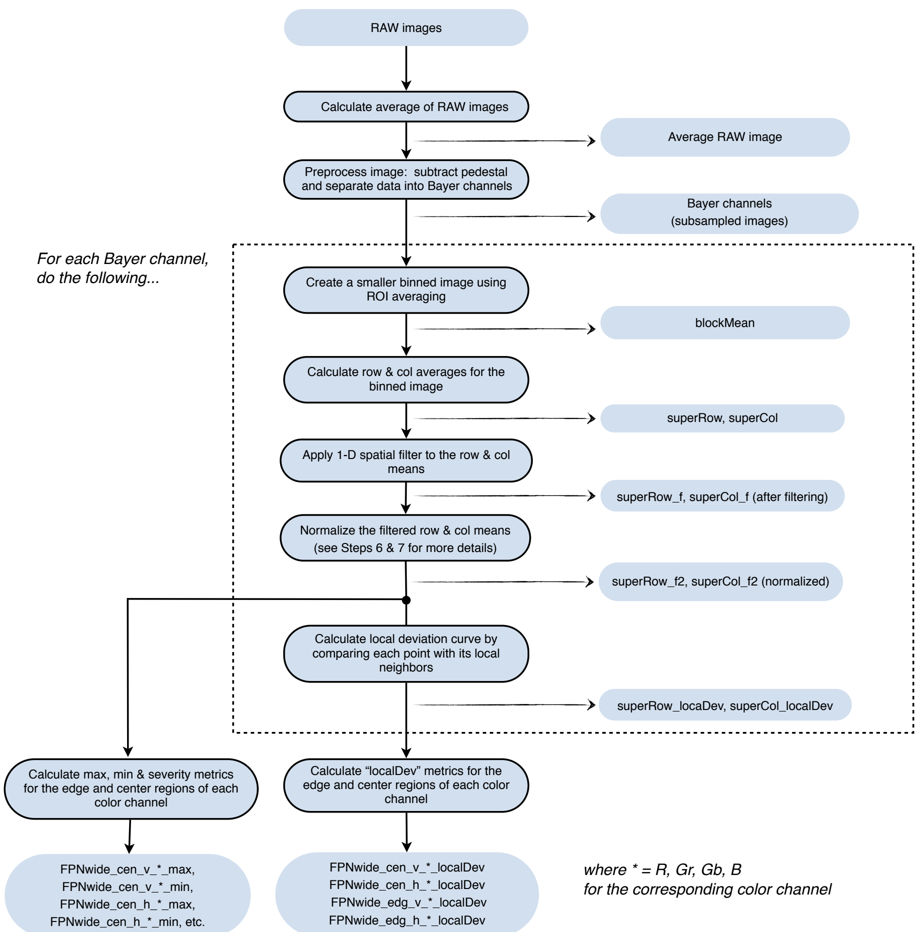


Figure 29: FPN-Wide Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
FPNwide_cen_h_*_max	max centre value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_edg_h_*_max	max edge value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_cen_v_*_max	max centre value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_edg_v_*_max	max edge value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_cen_h_*_min	min centre value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_edg_h_*_min	min edge value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_cen_v_*_min	min centre value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_edg_v_*_min	min edge value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_cen_h_*_localDev	max local deviation value (centre) of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_edg_h_*_localDev	max local deviation value (edge) of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_cen_v_*_localDev	max local deviation value (centre) of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_edg_v_*_localDev	max local deviation value (edge) of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	%
FPNwide_h_*_severity	severity value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B) (severity value = sum of the data points above the spec limits normalized by the length of the row)	%
FPNwide_v_*_severity	severity value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B) (severity value = sum of the data points above the spec limits normalized by the length of the column)	%
FPNwide_cen_h_*_max_x	the column number with max centre value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-

Name	Description	Uni t
FPNwide_cen_h_*_min_x	the column number with min centre value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-
FPNwide_edg_h_*_max_x	the column number with max edge value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-
FPNwide_edg_h_*_min_x	the column number with min edge value of the normalized filtered row means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-
FPNwide_cen_v_*_max_y	the row number with max centre value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-
FPNwide_cen_v_*_min_y	the row number with min centre value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-
FPNwide_edg_v_*_max_y	the row number with max edge value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-
FPNwide_edg_v_*_min_y	the row number with min edge value of the normalized filtered col means for a particular colour channel (i.e., replace * by R, Gr, Gb, or B)	-

Green Mismatch

The green channels may be mismatched in their signal level. The following test is used to identify abnormal lots and worst-case offenders.

Green Mismatch Setup

Test Hardware Setup

A D50 light source and an A light source, fitted with opal diffusers (as per Appendix A), should be used as the test targets.

Parameter	A	D50
Illuminant	Halogen	Filtered Halogen
Correlated Colour Temperature	2800K ± 20K	5000K ± 20K
Light level	300-800 cd/m	100-300 cd/m
Camera Distance from Light Source	8mm - 16mm	8mm - 16mm
Orientation	module up	

Module Hardware Setup

Module setup should follow module ERS default settings. In addition the settings below supersede any default settings:

Parameter	Value	Comment
Lens position	focused @ 2m	Lens position must be in a deterministic position for D50.
Software Driven AE	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog Gain	1x	
Integration Time	< 66 ms	Controlled by software driven AE, and limited by frame rate
Frame Rate	15 fps	
LSC	off	
Defect Pixel Correction	on	Refer to specific camera module ERS
White Balance	off	

Test Images

The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should be used to reduce temporal noise.

Test Image Specification	Value
Image captures	5 frames average
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

For this measurement, the subsampled Bayer image should be used for calculations. All subsequent calculations are to be done with double-precision math. Final output may be rounded to two decimal places.

1. Average 5 RAW images, captured under each of the test conditions outlined above.
2. Preprocess the images: subtract pedestal from each image, and separate the data into its four Bayer channels. Do not perform any of the other pre-processing steps. Bayer channel data should be used for all subsequent calculations.
3. Calculate the XY-coordinate locations for the centre ROI (cen) and the 4 corner ROIs: top left (TL), top right (TR), bottom left (BL), and bottom right (BR).
   ROI locations should be calculated using the formulas below, where the coordinates of the top-left pixel are (1,1) and the coordinates of the bottom-right pixel are (width, height).

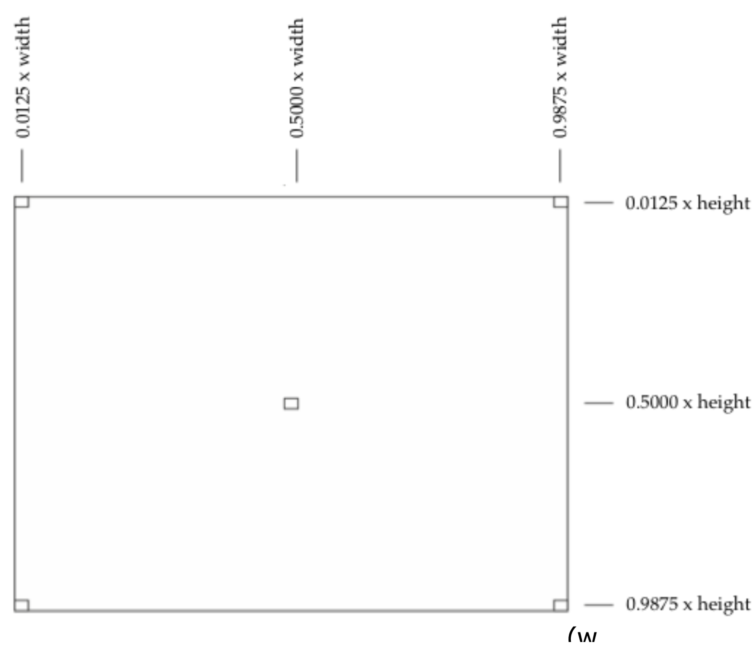


Figure 30 : Green Mismatch ROI's

XY-coordinates for the centre ROI:

ROIcentreX = 0.5w *ROIcentreY* = 0.5 h

XY-coordinates for the four corner ROIs:

ROIcentreX_TL = 0.0125 x w *ROIcentreY_TL* = 0.0125 h
ROIcentreX_TR = 0.9875 x w *ROIcentreY_TR* = 0.0125 h
ROIcentreX_BL = 0.0125 x w *ROIcentreY_BL* = 0.9875 h
ROIcentreX_BR = 0.9875 x w *ROIcentreY_BR* = 0.9875 h

where

w = horizontal image size in pixels
h = vertical image size in pixels

The size of the ROI (defined as a ratio of the total image size) should be 0.025

ROIsize = 0.025

The coordinates for the top left (x1, y1) and bottom right (x2, y2) pixel of each ROI should be calculated as follows:

x1 = Round (w * ROIcentreX - (w * ROIsize / 2) + 1.4999)
x2 = Round (w * ROIcentreX + (w * ROIsize / 2) - 0.5)
y1 = Round (h * ROIcentreY - (h * ROIsize / 2) + 1.4999)
y2 = Round (h * ROIcentreY + (h * ROIsize / 2) - 0.5)

where

ROIcentreX ,Y = location of the ROI, as a ratio of the total image size; range [0 1]
ROIsize = 0.025 = size of the ROI, as a ratio of the total image size; range [0 1]

The final ROI coordinates should all be rounded to the nearest integer.

4.Calculate the mean value for each corner ROI, as well as the centre ROI, and name these values

Gr_mean_TL, *Gr_mean_TR*, *Gr_mean_BL*, *Gr_mean_BR*, *Gr_mean_cen*
Gb_mean_TL, *Gb_mean_TR*, *Gb_mean_BL*, *Gb_mean_BR*, *Gb_mean_cen*

5.Calculate the green mismatch for each ROI, for both the D50 and A images, using the following equations:

greenMis_cen = 100 * | (*Gr_mean_cen* - *Gb_mean_cen*) | / ((*Gr_mean_cen* + *Gb_mean_cen*) / 2)

greenMis_BL = 100 * | (*Gr_mean_BL* - *Gb_mean_BL*) | / ((*Gr_mean_BL* + *Gb_mean_BL*) / 2)

etc.

Test Methodology Flow Chart

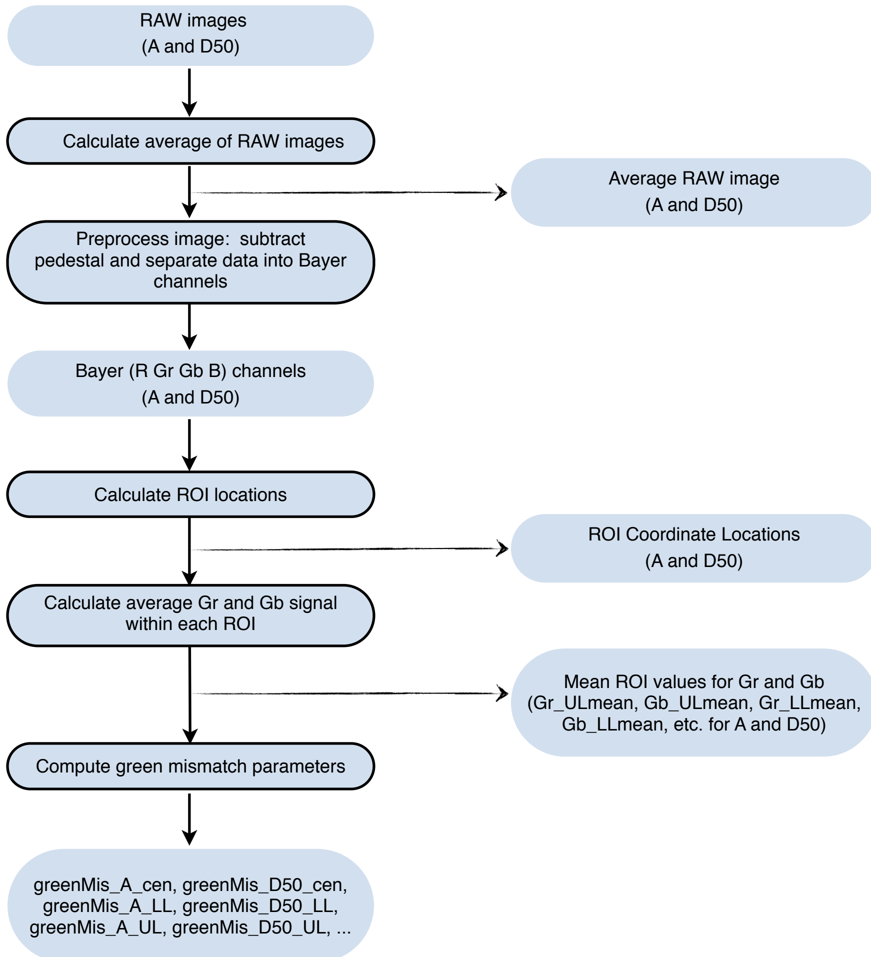


Figure 31 : Green Mismatch Test Methodology

Results Naming Convention in Log File

Name	Description	Unit
greenMis_D50_cen	Green Mismatch at image array centre for D50	%
greenMis_D50_TL	Green Mismatch at top left corner for D50	%
greenMis_D50_TR	Green Mismatch at top right corner for D50	%
greenMis_D50_BL	Green Mismatch at bottom left corner for D50	%
greenMis_D50_BR	Green Mismatch at bottom right corner for D50	%
greenMis_A_cen	Green Mismatch at image array centre for A light	%
greenMis_A_TL	Green Mismatch at top left corner for A light	%
greenMis_A_TR	Green Mismatch at top right corner for A light	%
greenMis_A_BL	Green Mismatch at bottom left corner for A light	%
greenMis_A_BR	Green Mismatch at bottom right corner for A light	%

Low Contrast Blemish

This document defines the LCB test setup and test methodology to be used for screening blemishes which are often caused by scratches, staining, or foreign material at any optical surface within the module. The flat field images should be captured and processed by the LCB detection algorithm described in the Test Algorithm section to calculate the blemish scores.

Low Contrast Blemish Setup

Test Hardware Setup

A D50 light source fitted with an opal diffuser (as per Appendix A) should be used as the test target.

Parameter	Value
Illuminant	D50
Correlated colour temperature	5000K ± 20K
Light level	100-300 cd/m
Target	Opal Diffuser
Camera distance to target	8 - 16mm
Orientation	module up

Module Hardware Setup

Module set up should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Lens position	focused @ 2m	Lens position must be in a deterministic position for D50.
Software driven AE	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog gain	1x	
Integration time	<66.67 ms	Controlled by SW exposure adjust, limited by frame rate
DPC	on	Refer to specific camera module ERS
Frame rate	15 fps	
LSC	off	
AWB	off	Manual white balance will be applied on test images

Test Image

The test images should be raw images captured at the maximum bit-depth (e.g. 10-bit). Manual white balance should be applied to ensure all colour planes achieve the same maximum response.

Test Image Specification	Value
Image captures	5 frames
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

Below are the image processing steps to derive the LCB scores.

1. Preprocess the images: subtract pedestal from each image, apply WB, and separate the data into its four Bayer channels. Do not perform any of the other pre-processing steps. Bayer channel data should be used for all subsequent calculations.

For each Bayer channel, perform the following operations:

2. Divide the Bayer channel into multiple ROIs, where the appropriate ROI size is defined by the module ERS (e.g., 9-by-9 Bayer pixels), and average the pixels in each ROI to form a binned version of the Bayer channel. In the case where the array size is not evenly divisible by the size of the ROI, the residual rows and columns should be evenly assigned to the border ROIs, resulting in border ROIs that may have a few more rows and/or columns than the other ROIs. Label the "scaled" image array (i.e., the binned Bayer channel) as follows:

    I_scaled

3. Pad I_scaled with n additional rows (or columns) on each edge via linear extrapolation using the first or last pair of data points (where n = (ROI size - 1)/2), and label the results as follows:

    I_scaled_pad_h - *I_scaled* with n extra padding columns
    I_scaled_pad_v - *I_scaled* with n extra padding rows

4. Apply a 1 x m (or m x 1, e.g., m = 3) 1-D median filter to the vertically (or horizontally) padded image, respectively.

    I_medfilt_h - vertical median filter applied to *I_scaled_pad_h*
    I_medfilt_v - horizontal median filter applied to *I_scaled_pad_v*

5. Perform 1-D spatial filtering on the horizontal (or vertical) direction respectively, with a 1-D filter kernel (h=[1/2, 0, ..., 0, -1, 0, ..., 1/2]), where the appropriate filter length is defined by the module ERS (e.g., 9 elements would correspond to h = [1/2, 0, 0, 0, -1, 0, 0, 0, 1/2]).

    I_filtered_pad_h - horizontal spatial filter applied to *I_medfilt_h*
    I_filtered_pad_v - vertical spatial filter applied to *I_medfilt_v*

6. Crop the array to remove the additional padded values that were added to the edges of the array. The array size should now match the original binned image size. Label these arrays as follows:

   I_filtered_h
   I_filtered_v

7. Clip all values in each array so that none of the values are lower than 0, nor higher than the full-scale deflection of the original image (e.g., for 10-bit images, the full scale value would be 1023).

8. Apply a 3x3 median filter to both the vertically-filtered image and the horizontally-filtered image.

9. Divide the filtered images into centre, edge, and corner regions, according to the diagram in Figure 29, where the width of the edge regions is equal to the filter length + 1.

10. Combine the filtered images *I_filtered_h* and *I_filtered_v* to form a new filtered image:

    I_filtered_c

    The rules for combining the images are as follows:

    For the centre region, use the average of the vertical and horizontal filtering results (i.e. average of *I_filtered_v* and *I_filtered_h*).

    For the left and right edges, use the vertical filtering result, *I_filtered_v*.
    For top and bottom edges, use the horizontal filtering result, *I_filtered_h*.
    For the 4 corner regions, use the minimum of the vertical and horizontal filtering results (i.e., minimum of I_filtered_v and I_filtered_h)

11. For any value in *I_filtered_h* or *I_filtered_v* that is larger than *threshold*, replace the value in *I_filtered_c* with the maximum value from either *I_filtered_h* or *I_filtered_v*, where *threshold* is an adjustable parameter that will be defined by Apple, based on control run data.

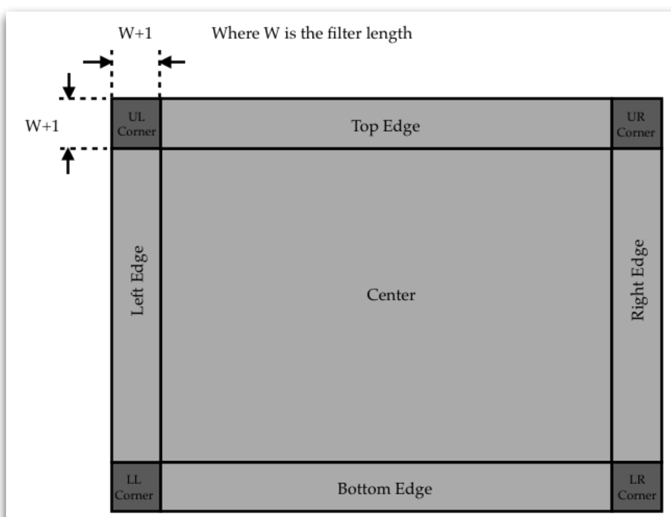


Figure 32: Partition of Corners, Edges, and Centre of an Image.

12. Calculate the following metrics based on the combined image, I_filtered_c:

LCB_centre	- the maximum pixel value of the centre region.
LCB_edge	- the maximum pixel value of the 4 edges.
LCB_corner	- the maximum pixel value of the 4 corners.

Also record the Bayer channel corresponding to each of these values, using the following letters to denote the different channels: R, Gr, Gb, B.

13. Record the (x,y) coordinates (indexed back to the full-resolution image) of centreMax, edgeMax, and cornerMax using the following variables:

LCB_centre_x, *LCB_centre_y*
LCB_edge_x, *LCB_edge_y*
LCB_corner_x, *LCB_corner_y*

Test Methodology Flow Chart

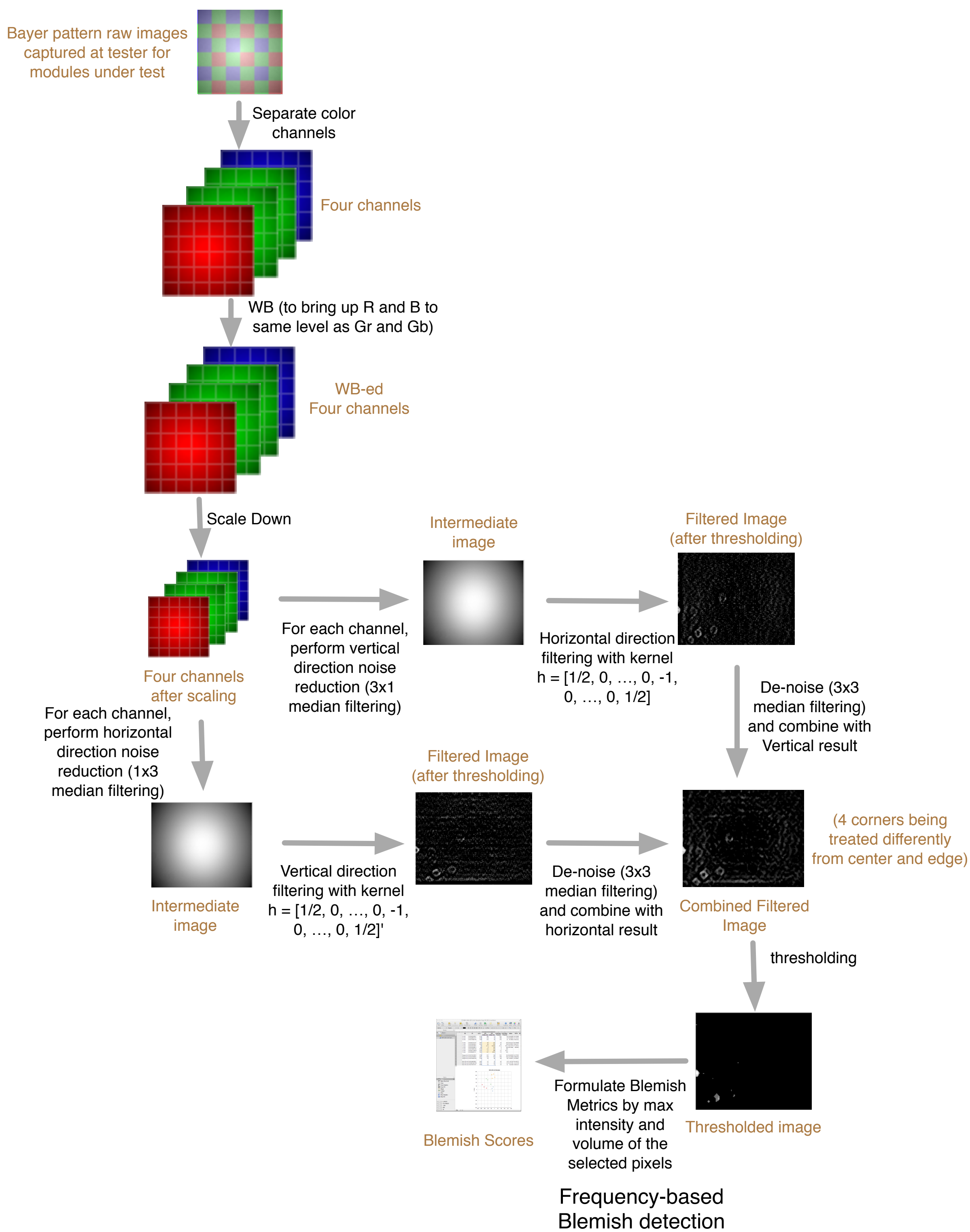


Figure 33: LCB test algorithm flow chart

Results Naming Convention in Log File

Name	Description	Unit
LCB_cen_max	max centre score of the 4 colour channels	LSB
LCB_cen_max_channel	the channel with the max centre score (R, Gr, Gb, B)	-
LCB_cen_max_x	x coordinate (indexed back to full-resolution image) with the max centre score	pixel
LCB_cen_max_y	y coordinate (indexed back to full-resolution image) with the max centre score	pixel
LCB_edge_max	max edge score of the 4 colour channels	LSB
LCB_edge_max_channel	the channel with the max edge score (R, Gr, Gb, B)	-
LCB_edge_max_x	x coordinate (indexed back to full-resolution image) with the max edge score	pixel
LCB_edge_max_y	y coordinate (indexed back to full-resolution image) with the max edge score	pixel
LCB_corner_max	max corner score of the 4 colour channels	LSB
LCB_corner_max_channel	the channel with the max corner score (R, Gr, Gb, B)	-
LCB_corner_max_x	x coordinate (indexed back to full-resolution image) with the max corner score	pixel
LCB_corner_max_y	y coordinate (indexed back to full-resolution image) with the max corner score	pixel
LCB_cen_R_max	max centre score of R channel	LSB
LCB_cen_Gr_max	max centre score of Gr channel	LSB
LCB_cen_Gb_max	max centre score of Gb channel	LSB
LCB_cen_B_max	max centre score of B channel	LSB
LCB_edge_R_max	max edge score of R channel	LSB
LCB_edge_Gr_max	max edge score of Gr channel	LSB
LCB_edge_Gb_max	max edge score of Gb channel	LSB
LCB_edge_B_max	max edge score of B channel	LSB
LCB_corner_R_max	max corner score of R channel	LSB
LCB_corner_Gr_max	max corner score of Gr channel	LSB
LCB_corner_Gb_max	max corner score of Gb channel	LSB

Name	Description	Unit
LCB_corner_B_max	max corner score of B channel	LSB

Low Contrast Blemish-Raw

This document defines the LCBraw test setup and test methodology to be used for screening blemishes which are often caused by scratches, staining, or foreign material at any optical surface within the module. The flat field images should be captured and processed by the LCBraw detection algorithm described in the Test Algorithm section to calculate the blemish scores.

Low Contrast Blemish Raw (LCBraw) Setup

Test Hardware Setup

A D50 light source fitted with an opal diffuser (as per Appendix A) should be used as the test target.

Parameter	Value
Illuminant	D50
Correlated colour temperature	5000K ± 20K
Light level	100-300 cd/m
Target	Opal Diffuser
Camera distance to target	8 - 16mm
Orientation	module up

Module Hardware Setup

Parameter	Value	Comments
Lens position	focused @ 2m	Lens position must be in a deterministic position for D50.
Software driven AE	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog gain	1x	
Integration time	<66.67 ms	Controlled by SW exposure adjust, limited by frame rate
DPC	on	Refer to specific camera module ERS
Frame rate	15 fps	
LSC	disabled	
AWB	disabled	Manual white balance will be applied on test images

Test Image

The test images should be raw images captured at the maximum bit-depth (e.g. 10-bit) . Manual white balance should be applied to ensure all colour planes achieve the same maximum response.

Test Image Specification	Value
Image captures	5 frames
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

Below are the image processing steps to derive the LCBraw scores.

1. Preprocess the images: subtract pedestal from each image, apply WB, and separate the data into its four Bayer channels. Do not perform any of the other pre-processing steps. Bayer channel data should be used for all subsequent calculations.
2. Divide the RAW image into multiple ROIs, where the appropriate ROI size is defined by the module ERS (e.g., 9-by-9 pixels), and average the pixels in each ROI to form a binned version of the RAW image. In the case where the array size is not evenly divisible by the size of the ROI, the residual rows and columns should be evenly assigned to the border ROIs, resulting in border ROIs that may have a few more rows and/or columns than the other ROIs. Label the "scaled" image array (i.e., the binned RAW image) as follows:

   I_scaled

3. Pad I_scaled with n additional rows (or columns) on each edge via linear extrapolation using the first or last pair of data points (where n = (ROI size - 1)/2), and label the results as follows:

   I_scaled_pad_h - *I_scaled* with n extra padding columns
   I_scaled_pad_v - *I_scaled* with n extra padding rows

4. Apply a 1 x m (or m x 1, e.g., m = 3) 1-D median filter to the vertically (or horizontally) padded image, respectively.

   I_medfilt_h - vertical median filter applied to *I_scaled_pad_h*
   I_medfilt_v - horizontal median filter applied to *I_scaled_pad_v*

5. Perform 1-D spatial filtering on the horizontal (or vertical) direction respectively, with a 1-D filter kernel (h=[1/2, 0, ..., 0, -1, 0, ..., 1/2]), where the appropriate filter length is defined by the module ERS (e.g., 9 elements would correspond to h = [1/2, 0, 0, 0, -1, 0, 0, 0, 1/2]).

   I_filtered_pad_h - horizontal spatial filter applied to *I_medfilt_h*
   I_filtered_pad_v - vertical spatial filter applied to *I_medfilt_v*

6. Crop the array to remove the additional padded values that were added to the edges of the array. The array size should now match the original binned image size. Label these arrays as follows:

    I_filtered_h
    I_filtered_v

7. Clip all values in each array so that none of the values are lower than 0, nor higher than the full-scale deflection of the original image (e.g., for 10-bit images, the full scale value would be 1023).

8. Apply a 3x3 median filter to both the vertically-filtered image and the horizontally-filtered image.

9. Divide the filtered images into centre, edge, and corner regions, according to the diagram in Figure 1, where the width of the edge regions is equal to the filter length + 1.

10. Combine the filtered images I_filtered_h and I_filtered_v to form a new filtered image:

    I_filtered_c

    The rules for combining the images are as follows:

    For the centre region, use the average of the vertical and horizontal filtering results (i.e. average of I_filtered_v and I_filtered_h).
    For the left and right edges, use the vertical filtering result, I_filtered_v.
    For top and bottom edges, use the horizontal filtering result, I_filtered_h.
    For the 4 corner regions, use the minimum of the vertical and horizontal filtering results (i.e., minimum of I_filtered_v and I_filtered_h)

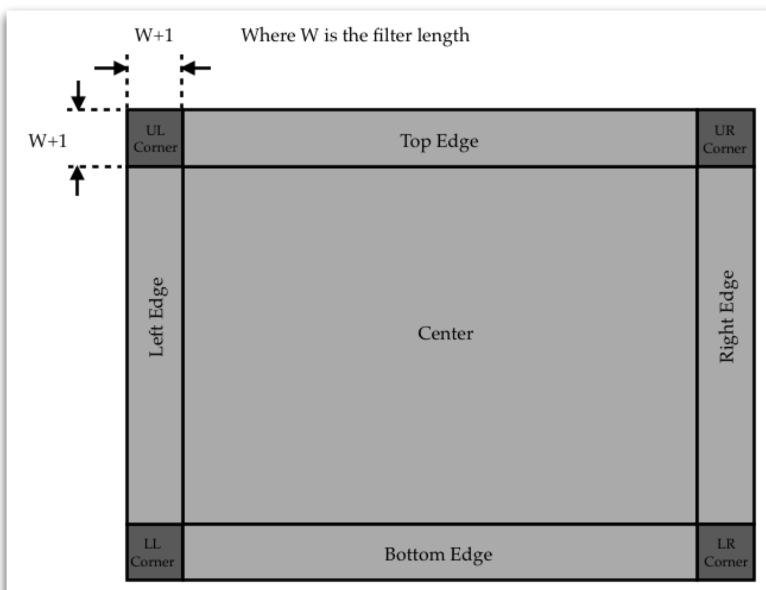


Figure 34: Partition of Corners, Edges, and Centre of an Image.

11. For any value in *I_filtered_h* or *I_filtered_v* that is larger than the *threshold*, replace the value in *I_filtered_c* with the original value from *I_filtered_h* or *I_filtered_v*, where severeFactor is an adjustable parameter that will be defined by Apple, based on control run data.

12. Calculate the following metrics based on the combined image, I_filtered_c:

    LCBraw_centre - the maximum pixel value of the centre region.
    LCBraw_edge - the maximum pixel value of the 4 edges.
    LCBraw_corner - the maximum pixel value of the 4 corners.

13. Record the (x,y) coordinates of centre max, edge max, and corner max using the following variables:

    LCBraw_centre_x, *LCBraw_centre_y*
    LCBraw_edge_x, *LCBraw_edge_y*
    LCBraw_corner_x, *LCBraw_corner_y*

Test Methodology Flow Chart

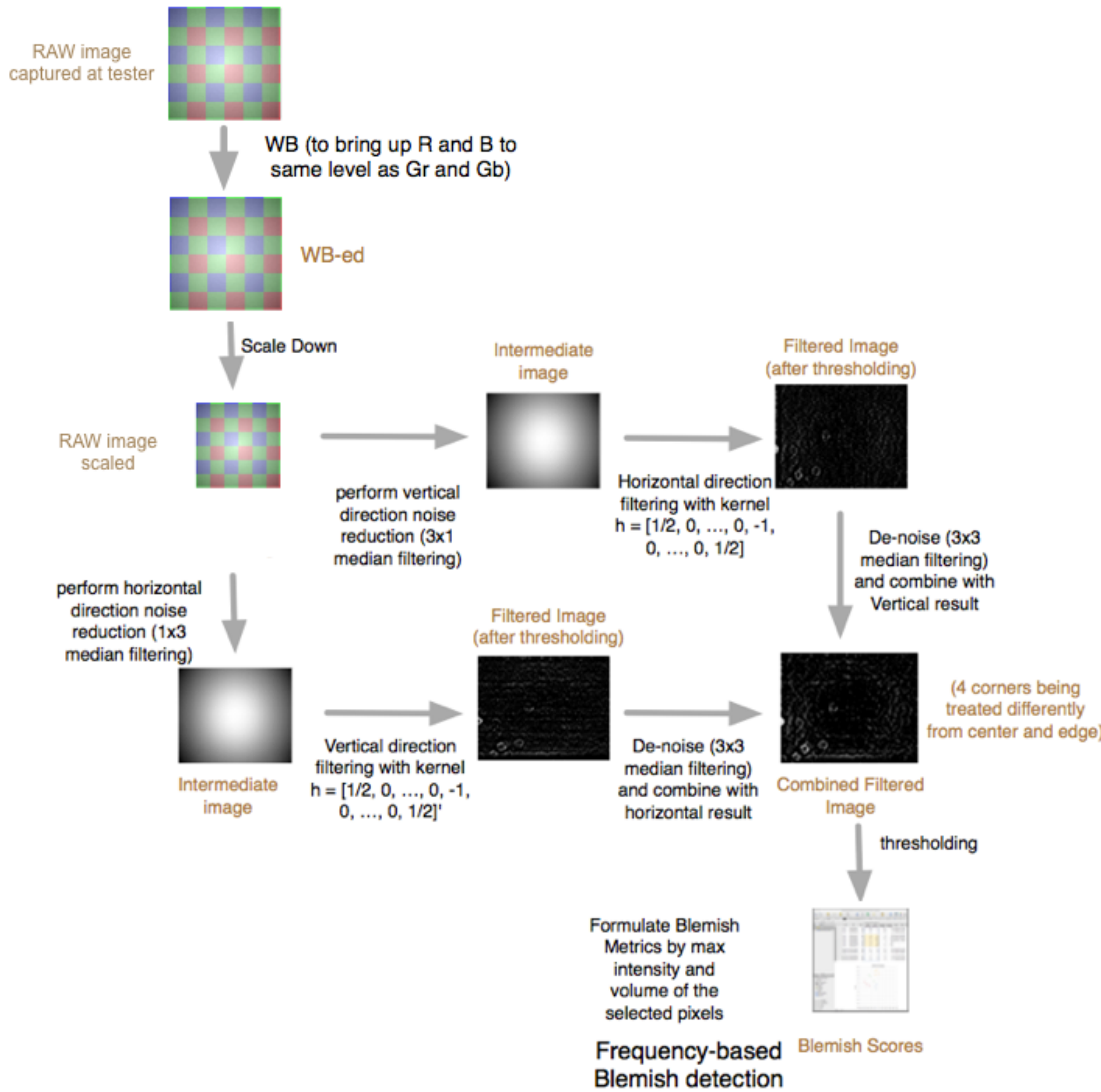


Figure 35 : LCBRaw Test Algorithm

Results Naming Convention in Log File

Name	Description	Unit
LCBraw_cen_max	max centre score	LSB
LCBraw_cen_max_x	the x coordinate with the max centre score	pixel
LCBraw_cen_max_y	the y coordinate with the max centre score	pixel
LCBraw_edge_max	max edge score	LSB
LCBraw_edge_max_x	the x coordinate with the max edge score	pixel
LCBraw_edge_max_y	the y coordinate with the max edge score	pixel
LCBraw_corner_max	max corner score	LSB
LCBraw_corner_max_x	the x coordinate with the max corner score	pixel
LCBraw_corner_max_y	the y coordinate with the max corner score	pixel

Optical Centre

Optical Centre Test Procedure

A light field created by passing light through an opal diffuser is used to calculate the optical centre of the module with respect to the sensor.

Test Hardware Setup

A D50 light source fitted with an opal diffuser (per Appendix A) should be used as the test target.

Parameter	Value
Illuminant	D50
Correlated Colour Temperature	5000K ± 20K
Light level	100-300 cd/m
Target	Opal Diffuser
Camera distance to target	8mm - 16mm
Orientation	module face up

Module Hardware Setup

Parameter	Value	Comments
Lens position	focused @ 2m	Lens position must be in a deterministic position for D50.
Software-driven AE	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog gain	>1x	
Integration time	<66.67ms	
DPC	on	see ERS for register settings
Frame rate	15fps	
LSC	off	
AWB	off	Manual white balance will be applied on test images

Test Images

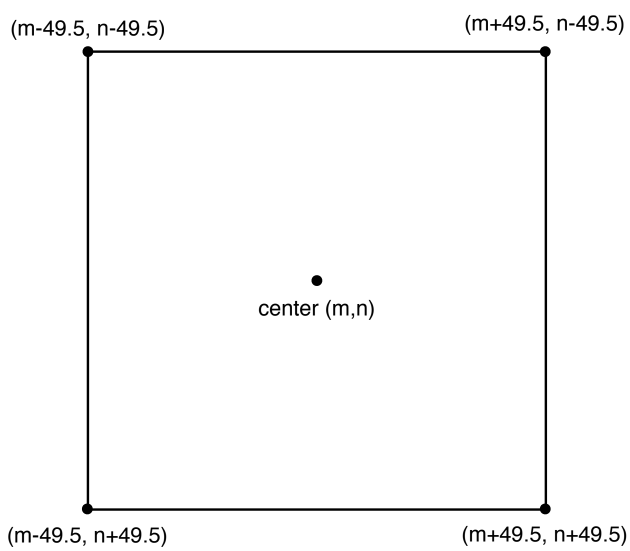
The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should be used to reduce temporal noise.

Test Image Specification	Value
Image captures	5 frames average
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

For this measurement, the full resolution luminance image should be used for calculations. All subsequent calculations are to be done with double-precision math. Final output should be rounded to one decimal place.

1. Average 5 RAW images, captured under the test conditions outlined above.
2. Preprocess the image (see "preprocess" document for more details): subtract pedestal from image, apply white balance, but do not apply lens shading correction, convert image from RGB to YCbCr. Output luminance channel (Y) should be used for all subsequent calculations.
3. Calculate the location of the ROIs, where each ROI is a 100 x 100 pixel region centreed at the ROI's centre coordinates. For example, if ROI centre = (m,n), then the four corners of the ROI are (m-49, n-49), (m-49, n+50), (m+50, n-49), (m+50, n+50)



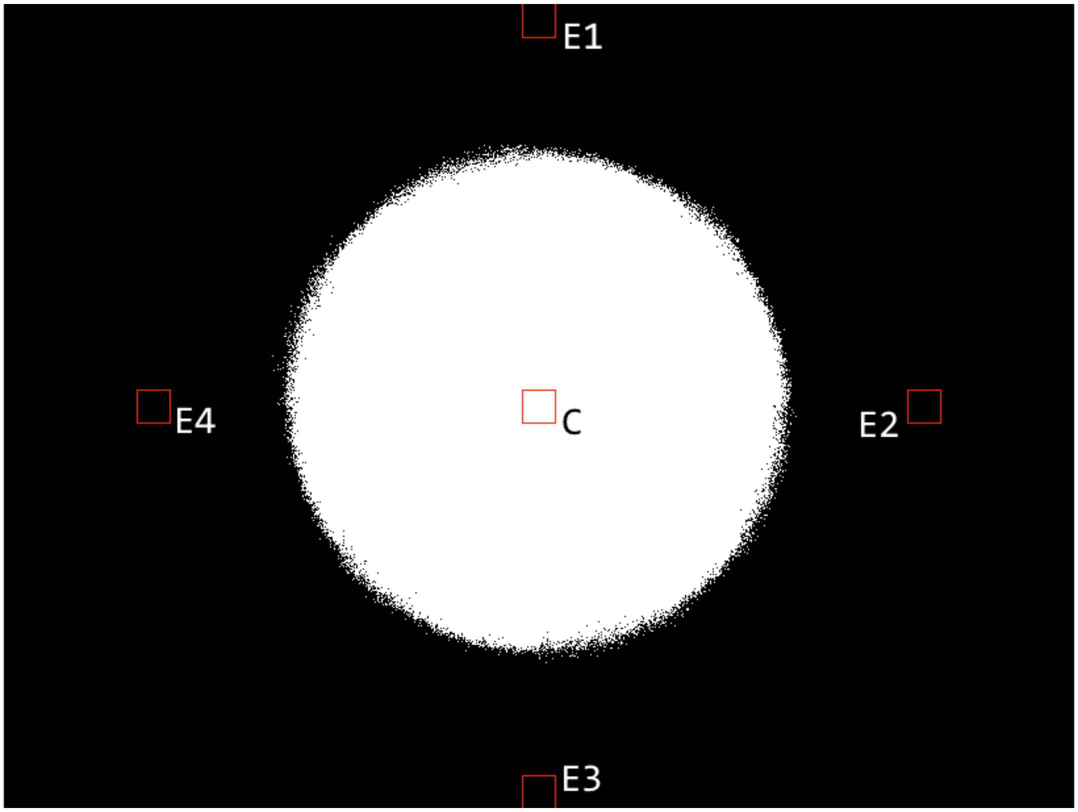


Figure 36 : ROI Coordinates for Optical Centre

The centre coordinates for the 5 ROI's are as follows:

centre C : (w/2+0.5, h/2+0.5)
top edge E1 : (w/2+0.5, 50.5)
right side E2 : (w/2+h/2-49.5, h/2+0.5)
bottom edge E3 : (w/2+0.5, h-49.5)
left side E4 : (w/2-h/2+50.5, h/2+0.5)

where

w = horizontal width of the image, in pixels
h = vertical height of the image, in pixels

Figure 34: Optical Centre ROI Coordinates Overlaid on an image

4. Calculate the average value for each ROI, and name the values as follows:

   meanC, *meanE1*, *meanE2*, *meanE3*, *meanE4*

5. Calculate the threshold value using the following formula:

$$threshold = \frac{meanC + \dfrac{meanE1 + meanE2 + meanE3 + meanE4}{4}}{2}$$

6. Use the threshold value to convert the image into a binary image, where pixel values above the threshold are set to 1 in the binary image, and pixel values below or equal to the threshold are set to 0.

7. Calculate the centroid (i.e., weighted average) of the central object in the binary image.

$$oc\_x = \frac{\sum B_i x_i}{\sum B_i}$$

$$oc\_y = \frac{\sum B_i y_i}{\sum B_i}$$

   Where *oc_x* and *oc_y* are the x- and y-coordinates of the centroid, Bi is the value of the ith pixel in the binary image, and *xi* and *yi* are the corresponding x- and y-coordinates of the ith pixel.

   Alternatively, an equivalent method of calculating the centroid would be to sum across each row and column, and then calculate the weighted average of the corresponding vector.

For example:

$$row\ (column)\ coordinates\ of\ centroid = \frac{\Sigma\, 1's\ per\ row\ (column) \times row\ (column)\ number}{total\ of\ 1's}$$

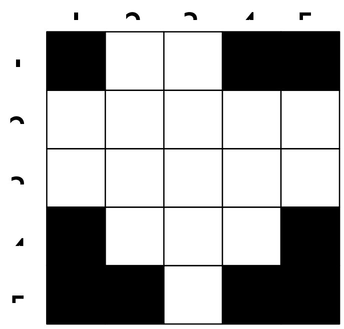


Figure 37: Centroid Sample for Calculation

$$row\ coordinates\ of\ centroid = \frac{2\times1 + 5\times2 + 5\times3 + 3\times4 + 1\times5}{2 + 5 + 5 + 3 + 1}$$

$$column\ coordinates\ of\ centroid = \frac{2\times1 + 4\times2 + 5\times3 + 3\times4 + 2\times5}{2 + 4 + 5 + 3 + 2}$$

Thus, the final x- and y-coordinates, rounded to the nearest tenth, would be oc_x = 2.9, oc_y = 2.8.

8. Report the row and column coordinates compared to the ideal absolute centre coordinates (w/2 + 0.5, h/2 + 0.5).

   oc_xShift = oc_x - (w/2 + 0.5)
   oc_yShift = (h/2 + 0.5) - oc_y

Note that the positive x and y directions are taken relative to Cartesian coordinates, with positive x being towards the right and positive y being towards the top.

Test Methodology Flow Chart

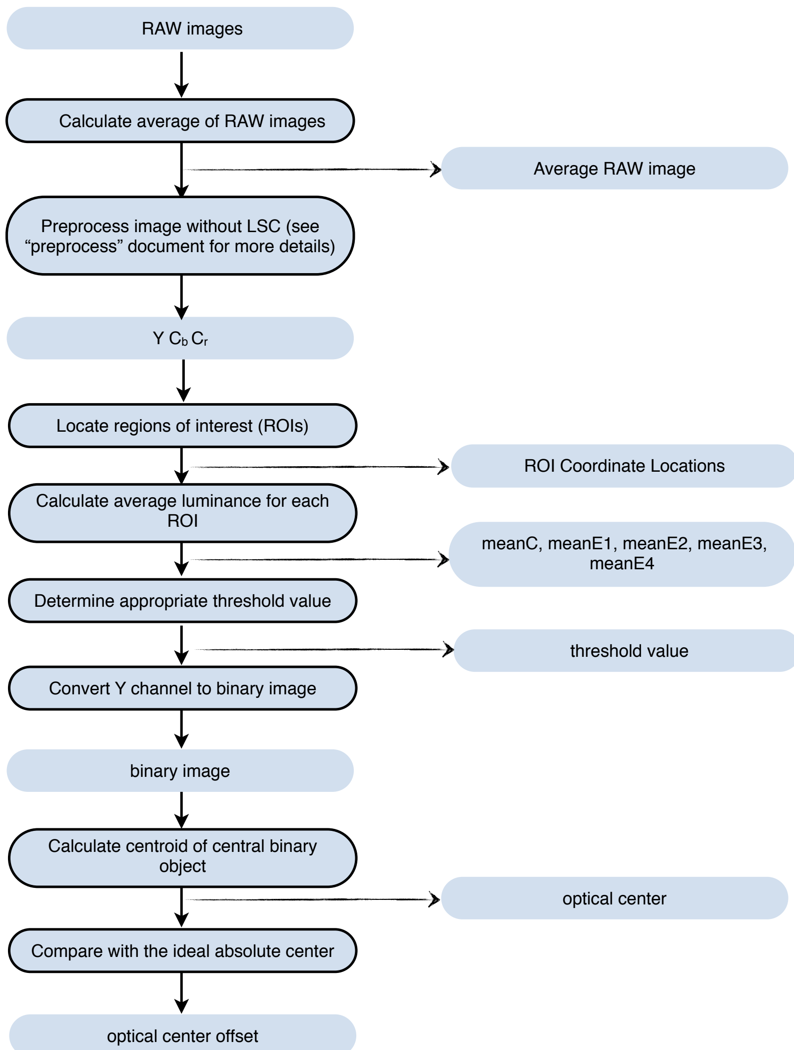


Figure 38 : Optical Centre Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
oc_threshold	Threshold value used to create binary image for calculating the optical centre	LSB
oc_x	x-coordinate of the optical centre	pix
oc_xShift	relative shift in the x-direction of the optical centre, with respect to the ideal centre coordinates	pix
oc_y	y-coordinate of the optical centre	pix
oc_yShift	relative shift in the y-direction of the optical centre, with respect to the ideal centre coordinates	pix

Pointing Tilt

Pointing tilt (tilt_x, tilt_y) measures the angular alignment between the ideal optical axis defined by the module datum and actual optical axis as defined by an image capture.

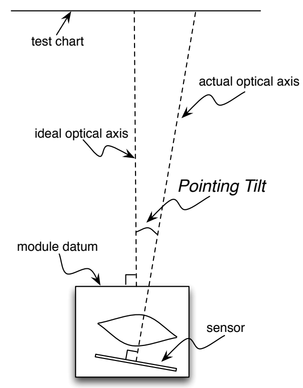


Figure 39 : Definition of Pointing Tilt

Pointing Tilt Setup

Test Hardware Setup

A transmissive Apple SFR chart with D65 light source should be used as the test target.

Parameter	Value
Illuminant	D65
Light Level	>200 cd/m
Target	high contrast transmissive Apple SFR target
Camera distance to target	Specific distances will be set by the camera module ERS. Use the same distances as those specified for measuring SFR. Setup distances must be measured with a calibrated laser distance meter
Orientation	Any orientation is acceptable

Test Chart

The test chart used for measuring pointing tilt is that same as that used for SFR testing. Please refer to the SFR test and SFR chart documentation for further details.

Module Hardware Setup

Module setup should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Software driven AE	70±5% FSD	average of a 64x64 pixel ROI around the white area which is left of the centre tilted square by 10% of the image width (refer to Figure 1) post pedestal subtraction, on y-channel
Analog gain	1x	
Integration time	<66.67 ms	controlled by AE, limited by frame rate
DPC	off	
Frame rate	15 fps	
LSC	enabled	
AWB	enabled	

Test Image

The test images should be full scale (e.g., 10-bit) RAW images captured.

Test Image Specification	Value
Image captures	1 frame
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

The tilt is calculation is based on a focused SFR image. The four inner field fiducial dots shown in figure 40 provide the basis for calculations. The field dot coordinates in image space have been detected to sub-pixel precision from an image of the SFR test chart. For details on detecting the fiducial dots, please refer to the SFR Test documentation. The final output is rounded to the nearest two decimal places.

Algorithm Notation:

chartDistance_mm : distance from camera to chart
fiducialDistanceInside_mm : distance between inside fiducials (mm)
actualDot : fiducial dots in image space (pixel)
actualDot_centre : centre of fiducial dots in image space (pixel)
idealDoti : ideal fiducial position in image space (pixel)

idealDot_centre : centre of ideal fiducial positions in image space (pixel)
convert_mmPerPixel : millimeter per pixel

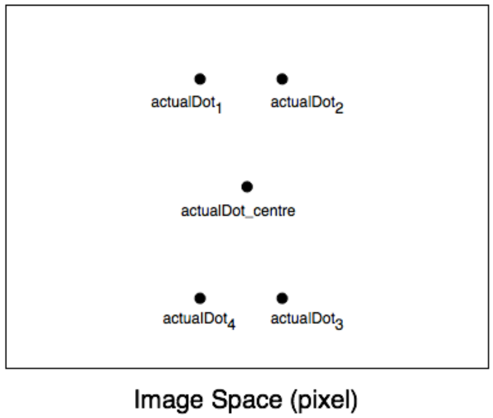


Figure 40: Four inner field fiducial dots on the Apple SFR chart in the Image Space

Detailed steps of the algorithm are outlined below, and a high level algorithm flow-chart is provided in Figure 41.

1. Preprocess the image (see "preprocess" document for more details): subtract pedestal from image, apply white balance & lens shading correction, apply demosaicing, and convert the image from RGB to YCbCr.

2. Calculate convert_mmPerPixel (MP)

   calculate values for two diagonals, two horizontals. and two verticals

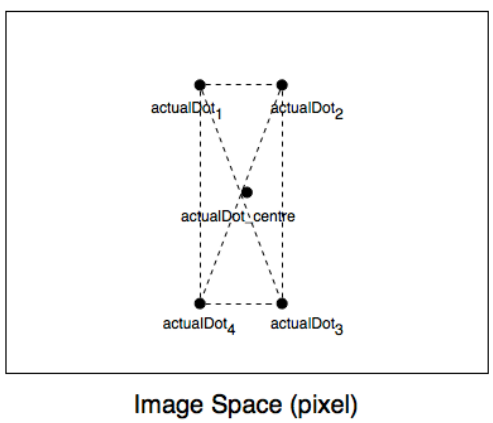


Figure 41: Lines used to determine conversion for mm per pixel

$$\text{convert\_mmPerPixel}_{diagonal} = \left|\frac{\text{physical distance}}{\text{image distance}}\right| = \frac{\sqrt{\text{fiducialDistance\_inside}_x^2 + \text{fiducialDistance\_inside}_y^2}}{\sqrt{\left(\text{actualDot}_{x(i)} - \text{actualDot}_{x(j)}\right)^2 + \left(\text{actualDot}_{y(i)} - \text{actualDot}_{y(j)}\right)^2}}$$

$$\text{convert\_mmPerPixel}_{horizontal} = \left|\frac{\text{physical distance}}{\text{image distance}}\right| = \frac{\text{fiducialDistance\_inside}_x}{\sqrt{\left(\text{actualDot}_{x(i)} - \text{actualDot}_{x(j)}\right)^2 + \left(\text{actualDot}_{y(i)} - \text{actualDot}_{y(j)}\right)^2}}$$

$$\text{convert\_mmPerPixel}_{vertical} = \left|\frac{\text{physical distance}}{\text{image distance}}\right| = \frac{\text{fiducialDistance\_inside}_y}{\sqrt{\left(\text{actualDot}_{x(i)} - \text{actualDot}_{x(j)}\right)^2 + \left(\text{actualDot}_{y(i)} - \text{actualDot}_{y(j)}\right)^2}}$$

Note the final value is taken as an average from the six lines.

3. Calculate centre of fiducial dots

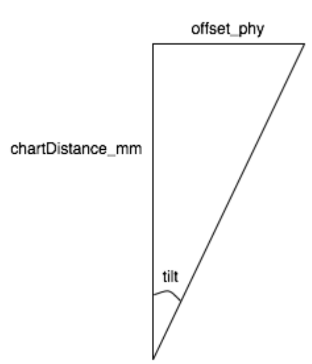
$$\text{actualDot\_centre} = (\text{actualDot\_centre}_x, \text{actualDot\_centre}_y) = \frac{\sum \text{actualDot}_i}{\#\text{ of dots}} = \left(\frac{\sum \text{actualDot}_{x(i)}}{4}, \frac{\sum \text{actualDot}_{y(i)}}{4}\right)$$

4. Calculate the distance between calculated centre and ideal centre in physical space

$$\text{idealDot\_centre} = \left(\frac{\text{w}}{2} + 0.5, \frac{\text{h}}{2} + 0.5\right)$$

$$\begin{aligned}\text{offset\_phy} &= \text{convert\_mmPerPixel} \cdot (\text{actualDot\_centre} - \text{idealDot\_centre}) \\ &= \text{convert\_mmPerPixel} \cdot \left(\text{actualDot\_centre}_x - \text{idealDot\_centre}_x, \text{actualDot\_centre}_y - \text{idealDot\_centre}_y\right)\end{aligned}$$

5. Calculate the angle of the pointing tilt



$$\begin{aligned}\text{tilt} &= (\text{tilt}_x, \textit{tilt}_y) \\ &= \arctan\left(\frac{\text{offset\_phy}}{\text{chartDistance\_mm}}\right) \\ &= \left(\arctan\left(\frac{\text{offset\_phy}_x}{\text{chartDistance\_mm}}\right), \arctan\left(\frac{\text{offset\_phy}_y}{\text{chartDistance\_mm}}\right)\right)\end{aligned}$$

Figure 42 : Calculating the angle of the Pointing Tilt

Note that the positive x and y directions are taken relative to Cartesian coordinates, with positive x being towards the right and positive y being towards the top.

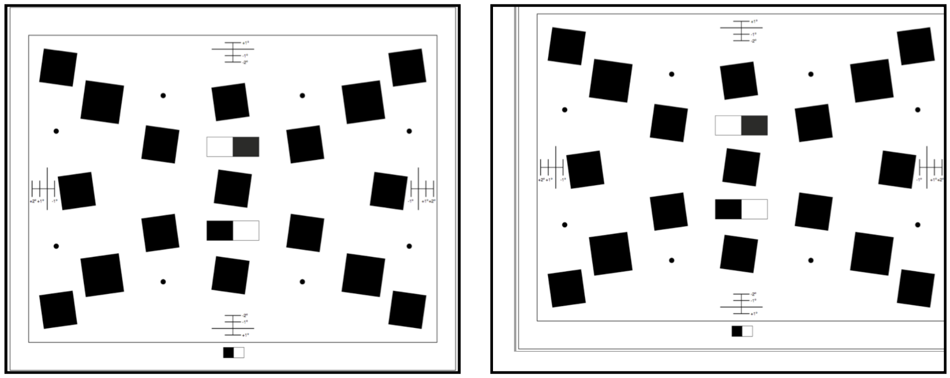


Figure 43: Diagram of (a) no shift and (b) positive x- and y-pointing tilt

Test Methodology Flow Chart

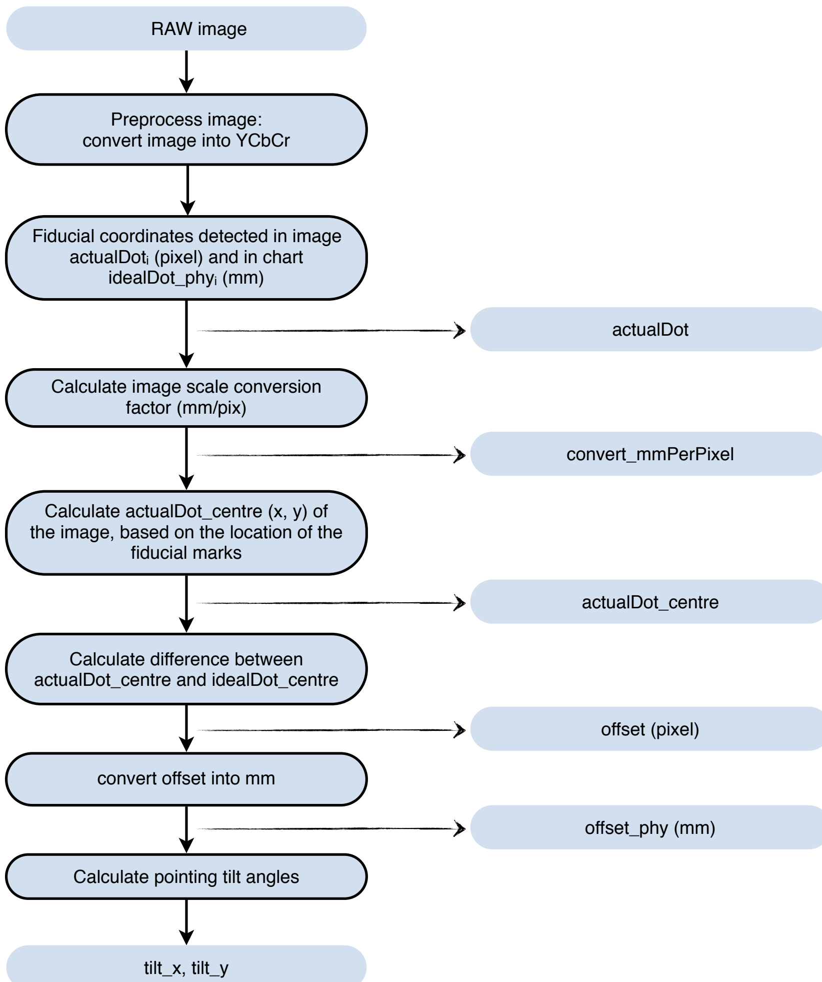


Figure 44 : Pointing Tilt Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
alignment_tilt_x	module x-tilt with respect to ideal optical axis	°
alignment_tilt_y	module y-tilt with respect to ideal optical axis	°

Read Noise

This document provides a common metric for characterizing the read noise of the sensor.

Read Noise Test Setup

Test Hardware Setup

Parameter	Value
Illuminant	Dark
Light level	<1 lux

Module Hardware Setup

Parameter	Value	Comments
Analog Gain	max gain	
Integration Time	<1ms	Use the minimum integration time available for the sensor
DPC	on	
Frame Rate	30fps	
LSC	off	
AWB	off	

Test Images

The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should not be used

Test Image Specification	Value
Image captures	2 single frame captures
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

For this measurement, the full resolution RAW image should be used for calculations. All subsequent calculations are to be done with double-precision math. Final output may be rounded to two decimal places.

1. Capture 2 RAW images, raw1 and raw2, under the test conditions outlined above.

2. Subtract image 1 from image 2.
   Note: the result should be a signed value, since some of the numbers may be negative.)

   raw_delta = raw2 - raw1

3. Apply the following equation to calculate read noise:

   noise_dark_read = stdev(raw_delta) / √2

   where stdev( ) is the standard deviation over all pixels in the array.

Test Methodology Flow Chart

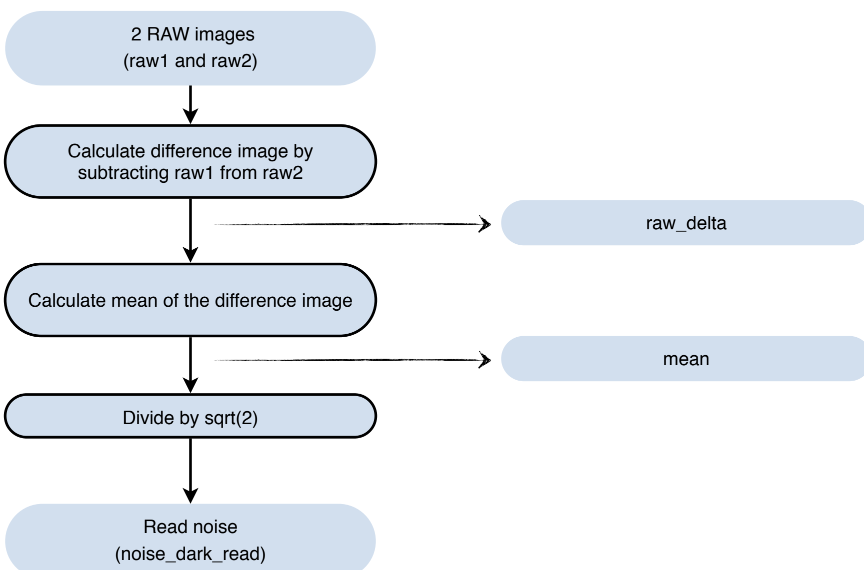


Figure 45 : Read Noise Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
noise_dark_read	Sensor read noise	LSB

Relative Illumination

A light field created by passing light through an opal diffuser is used to calculate relative illumination.

Relative Illumination Setup

Test Hardware Setup

A D50 light source fitted with an opal diffuser (per Appendix A) should be used as the test target.

Parameter	Value
Illuminant	D50
Correlated colour temperature	5000K ± 20K
Light level	100-300 cd/m
Target	diffuser
Camera distance to target	8 - 16mm
Orientation	module up

Module Hardware Setup

Parameter	Value	Comments
Lens position	focused @ 2m	Lens position must be in a deterministic position for D50.
SW Exposure Adjust	80±5% FSD	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
Analog Gain	1x	
Integration Time	<66.67ms	Controlled by SW exposure adjust to meet LSB target, limited by frame rate
DPC	on	
Frame Rate	15fps	
LSC	off	
AWB	off	Manual white balance will be applied on test images

Test Images

The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should be used to reduce temporal noise.

Test Image Specification	Value
Image captures	5 frames average
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

For this measurement, the full resolution luminance image should be used for calculations. All subsequent calculations are to be done with double-precision math. Final output may be rounded to two decimal places.

1. Average 5 RAW images, captured under the test conditions outlined above.

2. Preprocess the image (see “preprocess” document for more details): subtract pedestal from image, apply white balance, but do not apply lens shading correction, convert image from RGB to YCbCr. Output luminance channel (Y) should be used for all subsequent calculations.

3. Calculate the XY-coordinate locations for the centre ROI (cen) and the 4 corner ROIs: upper left (UL), upper right (UR), lower left (LL), lower right (LR).

   ROI locations should be calculated using the formulas below, where the coordinates of the top-left pixel are (1,1) and the coordinates of the bottom-right pixel are (width, height).

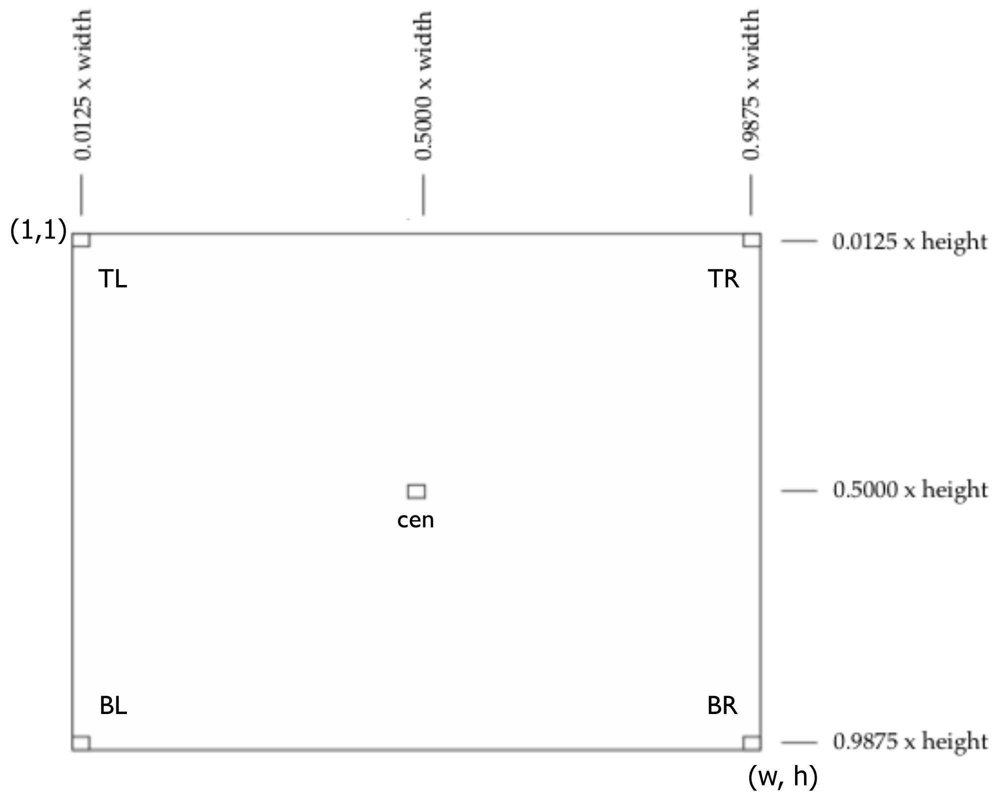


Figure 46: Diagram of ROI locations for relative illumination calculations

XY-coordinates for the centre ROI:

ROIcentreX = 0.5 x w   *ROIcentreY* = 0.5 x h

XY-coordinates for the four corner ROIs:

ROIcentreX_TL = 0.0125 x w   *ROIcentreY_TL* = 0.0125 x h
ROIcentreX_TR = 0.9875 x w   *ROIcentreY_TR* = 0.0125 x h
ROIcentreX_BL = 0.0125 x w   *ROIcentreY_BL* = 0.9875 x h
ROIcentreX_BR = 0.9875 x w   *ROIcentreY_BR* = 0.9875 x h

where
w = horizontal image size in pixels
h = vertical image size in pixels

The size of the ROI (defined as a ratio of the total image size) should be 0.025

ROIsize = 0.025

The coordinates for the top left (x1, y1) and bottom right (x2, y2) pixel of each ROI should be calculated as follows:

$x1$ = Round (w * $ROIcentreX$ - (w * $ROIsize$ / 2) + 1.4999)
$x2$ = Round (w * $ROIcentreX$ + (w * $ROIsize$ / 2) - 0.5)
$y1$ = Round (h * $ROIcentreY$ - (h * $ROIsize$ / 2) + 1.4999)
$y2$ = Round (h * $ROIcentreY$ + (h * $ROIsize$ / 2) - 0.5)

where

ROIcentreX ,Y = location of the ROI, as a ratio of the total image size; range [0 1]
ROIsize = 0.025 = size of the ROI, as a ratio of the total image size; range [0 1]

The final ROI coordinates should all be rounded to the nearest integer.

4. Calculate the mean value for each corner ROI, as well as the centre ROI:

   ULmean
   URmean
   LLmean
   LRmean
   cenMean

5. Compute the relative illumination parameters.

   Relative illumination is calculated by dividing the average value of the ROI in each corner of the image by the average value of the ROI in the centre of the image. The lowest relative illumination value for each corner is compared with the specification limit, set by Apple.

   ri_UL_ratio = 100 * *ri_UL_mean* / *ri_cen_mean*
   ri_UR_ratio = 100 * *ri_UR_mean* / *ri_cen_mean*
   ri_LL_ratio = 100 * *ri_LL_mean* / *ri_cen_mean*
   ri_LR_ratio = 100 * *ri_LR_mean* / *ri_cen_mean*

6. Compute the maximum relative illumination delta

7. Relative illumination delta is the difference in relative illumination between the brightest and the dimmest corner. In other words, the difference between the maximum and minimum of *ri_UL_ratio*, *ri_UR_ratio*, *ri_LL_ratio*, *ri_LR_ratio* is the RI delta.

ri_delta = (max(*ri_UL_ratio*, *ri_UR_ratio*, *ri_LL_ratio*, *ri_LR_ratio*)
   - min(*ri_UL_ratio*, *ri_UR_ratio*, *ri_LL_ratio*, *ri_LR_ratio*))

Test Methodology Flow Chart

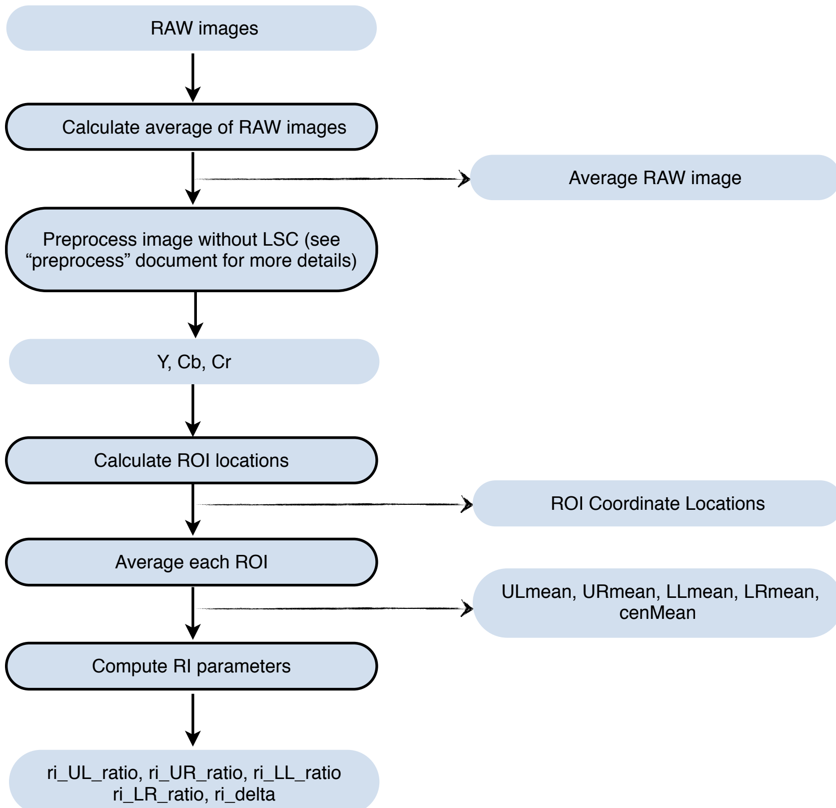


Figure 47 : Relative Illumination Test Procedure

Results Naming Convention in Log File

Name	Description	Unit
ri_TL_ratio	Relative illumination of upper right corner compared to centre ROI	%
ri_TR_ratio	Relative illumination of upper right corner compared to centre ROI	%
ri_BL_ratio	Relative illumination of lower left corner compared to centre ROI	%
ri_BR_ratio	Relative illumination of lower right corner compared to centre ROI	%
ri_delta	Difference between max and min of 4 corners' Relative Illumination	%

Relative Uniformity

This document defines the relative uniformity (RU) test setup and test methodology to be used for screening radial colour non-uniformity caused by: 1) abnormal colour shading behaviour 2) diffused colour blemish caused by the defect of image sensor response or IR cut filter transmittance.The flat field images shall be captured and processed by the relative uniformity algorithm described in Test Methodology.

Relative Uniformity Setup

Test Hardware Setup

A D50 light source (as per Appendix A) should be used as the test target.

	value
Illuminant	D50
Colour temperature	5000K ± 20K
Light level	100-300 cd/m
Target	Opal diffuser
Distance to target	8 - 16mm
Orientation	module up

Module Hardware Setup

	value	comments
Lens position	focused @ 2m	Lens position must be in a deterministic position for D50.
Software driven AE	80±5% FSD	target centre 100x100 pixels of highest colour plane adjusted by integration time
Analog gain	1x	
Integration time	<66.67ms	controlled by software AE to meet LSB target, limited by frame rate
DPC	on	as per module ERS
Frame rate	15fps	
LSC	off	
AWB	off	

Test Images

The test images shall be averaged Bayer images in 10-bit captured with module LSC and AWB off.

Test Image Specification	
Image Captures	5 frames average
Image Bit-depth	Full scale bit-depth (e.g., 10-bit)
Image Format	Bayer

Test Methodology

Preface: All subsequent calculations are to be done with signed double-precision math. Final output may be rounded to two decimal places.

1. Subtract pedestal from averaged Raw Image.

2. Separate into Bayer Channels

3. Define ROIs

   The input data array is divided into multiple ROIs with each ROI radially tiled from the centre defined ROI.

   Each ROI is a square where the side length = 0.025 x image pixel diagonal / sqrt(2)

   ROI rings are to be placed radially out from the image centre. Each ring extending our from the centre, R0% to R95% are to be noted as their percentage field position away from the centre.

   ROI rings are to be located at every multiple of 0.025 x image pixel diagonal from image centre to image corners.

   (Sj) = 0.025 x image pixel diagonal

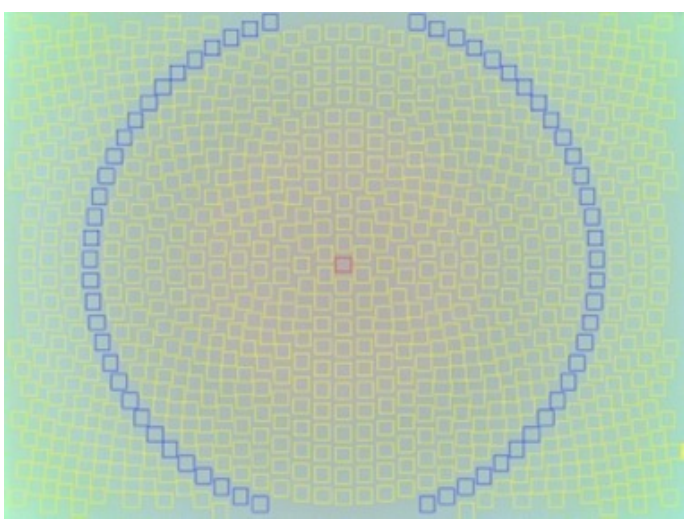
-


Figure 48: Example ROIs plotted on a image with a radial colour shading problem. The centre ROI of the image is highlighted in red. ROIs at 60%(R60%)of field are highlighted in blue.

4. Application of metrics

The following metrics below should be applied to Channels as well as specified ratios. The ratios should be calculated from the ROI averages and then put into the metrics calculations.

Apply metrics to the following specifications:

RD from R10% to R90%

Red, GreenR, GreenB, Blue, Red / Blue ratio, Red / Gr ratio, Gr / Gb ratio.

FND from R10% to R90%

Red, GreenR, GreenB, Blue, Red / Blue ratio, Red / Gr ratio, Gr / Gb ratio.

fieldRange from R10% to R90%

Red, GreenR, GreenB, Blue, Red / Blue ratio, Red / Gr ratio, Gr / Gb ratio.

5. Calculate the follow metrics from the ROI defined colour channel and/or ratio.

The Ring Difference %: Maximum percent difference from R i from its previous two rings.

predicted Pi = avg(RRing i-1) + (avg(RRing i-1) - avg (RRing i-2))

RD = min/maximage((avg (RRing i) - Pi) / Pi) x 100

The Field Neighbor difference %: Maximum percentage difference between neighboring rings over all rings.

FND = maximage (| (RROI i - RROI i-1) |) * 100

The Field Difference Max %: Maximum percentage difference between ROIs in the same ring over all ROIs in the ring.

fieldRange = maximage ((maxRing i - minRing i) / avg (RRing i))) * 100

Test Methodology Flow Chart

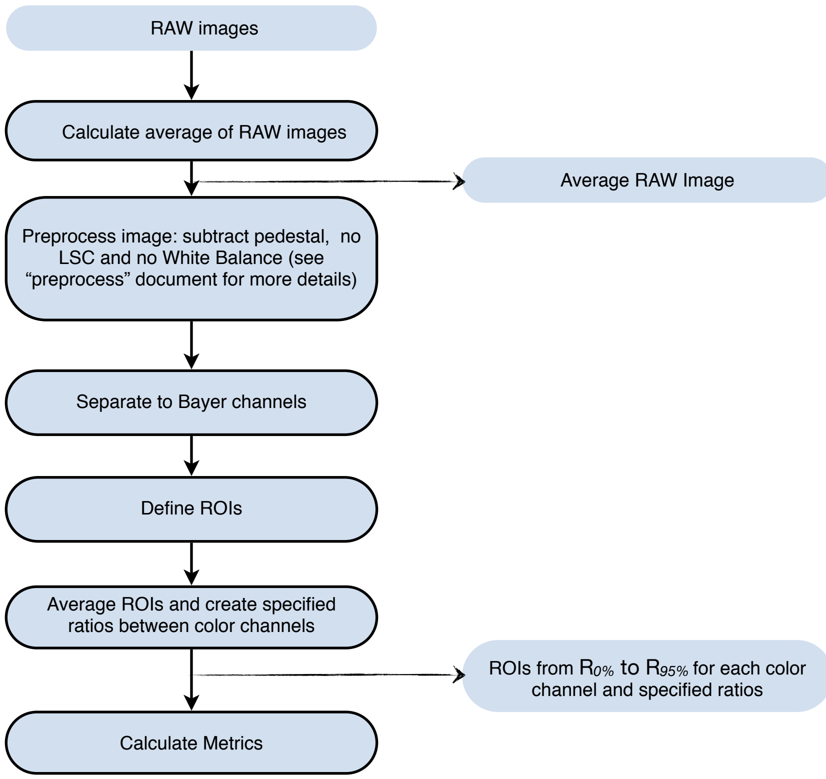


Figure 49 : Relative Uniformity Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
ru_rd_R_max	Ring Difference max from R	%
ru_rd_Gr_max	Ring Difference max from R	%
ru_rd_Gb_max	Ring Difference max from R	%
ru_rd_B_max	Ring Difference max from R	%
ru_rd_RB_max	Ring Difference max from R	%
ru_rd_RGr_max	Ring Difference max from R Ratio	%
ru_rd_GrGb_max	Ring Difference max from R Ratio	%

Name	Description	Unit
ru_rd_R_min	Ring Difference min from R	%
ru_rd_Gr_min	Ring Difference min from R	%
ru_rd_Gb_min	Ring Difference min from R	%
ru_rd_B_min	Ring Difference min from R	%
ru_rd_RB_min	Ring Difference min from R	%
ru_rd_RGr_min	Ring Difference min from R - tio	%
ru_rd_GrGb_min	Ring Difference min from R Ratio	%
ru_fnd_R_max	Field Neighbor Difference max from R channel	%
ru_fnd_Gr_max	Field Neighbor Difference max from R GreenR channel	%
ru_fnd_Gb_max	Field Neighbor Difference max from R Greenb channel	%
ru_fnd_B_max	Field Neighbor Difference max from R channel	%
ru_fnd_RB_max	Field Neighbor Difference max from R Blue Ratio	%
ru_fnd_RGr_max	Field Neighbor Difference max from R GreenR Ratio	%
ru_fnd_GrGb_max	Field Neighbor Difference max from R GreenR/GreenB Ratio	%
ru_fieldRange_R_max	Field Range max from R	%
ru_fieldRange_Gr_max	Field Range max from R	%
ru_fieldRange_Gb_max	Field Range max from R	%
ru_fieldRange_B_max	Field Range max from R	%
ru_fieldRange_RB_max	Field Difference max from R	%
ru_fieldRange_RGr_max	Field Difference max from R Ratio	%
ru_fieldRange_GrGb_max	Field Difference max from R Ratio	%

Rotation

Rotation Test Setup

Test Hardware Setup

A transmissive Apple SFR chart with D65 light source should be used as the test target.

Parameter	Value
Illuminant	D65
Light Level	>200 cd/m
Target	high contrast transmissive Apple SFR target
Camera distance to target	Specific distances will be set by the camera module ERS. Use the same distances as those specified for measuring SFR. Setup distances must be measured with a calibrated laser distance meter
Relay lens	Relay lenses (if any) will be specified in the camera module ERS
Orientation	Any orientation is acceptable

Test Chart

The test chart used for measuring pointing tilt is that same as that used for SFR testing. Please refer to the SFR test and SFR chart documentation for further details.

Module Hardware Setup

Module setup should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Software driven AE	70±5% FSD	average of a 64x64 pixel ROI around the white area which is left of the centre tilted square by 10% of the image width (refer to Figure 1) post pedestal subtraction, on y-channel
Analog gain	1x	
Integration time	<66.67 ms	controlled by AE, limited by frame rate
DPC	off	
Frame rate	15 fps	
LSC	off	
AWB	off	

Test Image

The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should be used to reduce temporal noise.

Test Image Specification	Value
Image captures	1 frame
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

The rotation is calculation is based on a focused SFR image. The four outer field fiducial dots shown in figure 2 provide the basis for calculations. The field dot coordinates in image space have been detected to sub-pixel precision from an image of the SFR test chart. For details on detecting the fiducial dots, please refer to the SFR Test documentation. The final output is rounded to the nearest two decimal places.

Algorithm Notation:

actualDot : fiducial dots in image space (pixel)
actualDot_centre : centre of fiducial mark in image space (pixel)
idealDot : fiducial dots in physical space (mm)

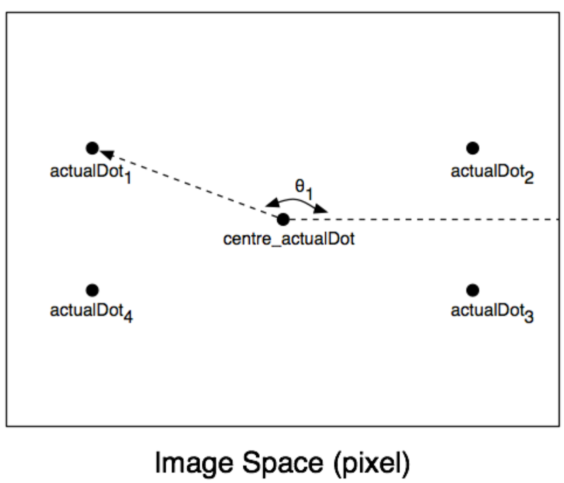


Figure 50: Four outer field fiducial dots on the Apple SFR chart in the Image Space

Detailed steps of the algorithm are outlined below, and a high level algorithm flow-chart is provided in Figure 50.

1. Preprocess the image: subtract pedestal from image, apply white balance & lens shading correction, apply demosaicing, and convert the image from RGB to YCbCr.

2. Calculate the centre of the fiducial dots

$$\begin{aligned}\text{actualDot\_centre} &= (\text{actualDot\_centre}_x, \text{actualDot\_centre}_y) \\ &= \frac{\sum \text{actualDot}_i}{\#\text{ of dots}} \\ &= \left(\frac{\sum \text{actualDot}_{x(i)}}{4}, \frac{\sum \text{actualDot}_{y(i)}}{4}\right)\end{aligned}$$

3. Calculate the cartesian angle of the vector from actualDot_centre to each fiducial dot

$$\vartheta_i = \arctan\left(\frac{\text{actualDot}_{x(i)} - \text{actualDot\_centre}_x}{\text{actualDot}_{y(i)} - \text{actualDot\_centre}_y}\right)$$

4. Calculate the cartesian angle of the vector from idealDot_centre to each ideal fiducial dot

$$\Theta_i = \arctan\left(\frac{\text{idealDot}_{x(i)} - \text{idealDot\_centre}_x}{\text{idealDot}_{y(i)} - \text{idealDot\_centre}_y}\right)$$

5. Subtract the actual rotation from ideal rotation and average all values. Report final result.

$$\Delta\vartheta_i = \vartheta_i - \Theta_i$$

$$\Delta\vartheta = \frac{\sum \Delta\vartheta_i}{4}$$

Note that the positive rotation directions are taken relative to Cartesian coordinates, with positive rotation being counterclockwise.

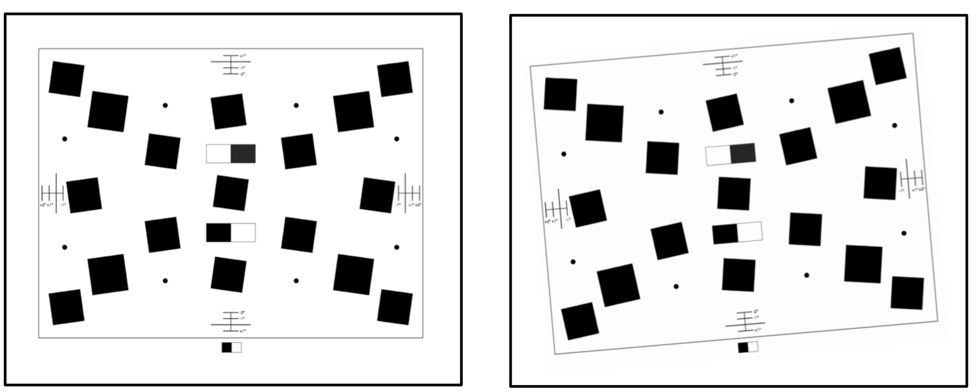


Figure 51: *Diagram of (a) no shift and (b) positive rotation*

Test Methodology Flow Chart

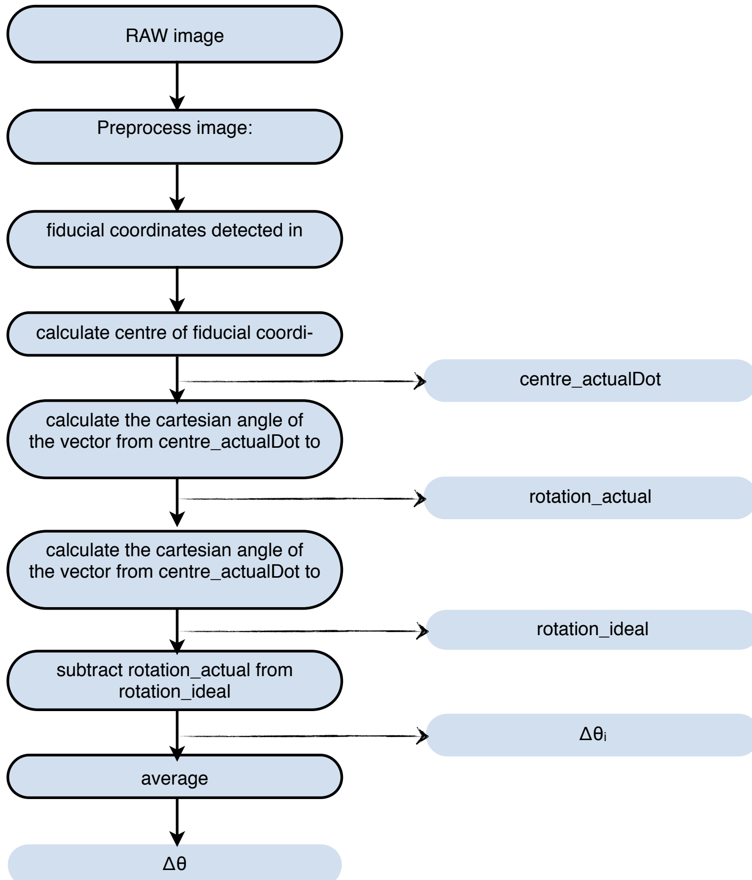


Figure 52: Rotation Test Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
alignment_rotation	Camera module rotational alignment	°

Spatial Frequency Response

The Spatial Frequency Response (SFR) test measures the resolution/sharpness/modulation transfer function at various field points of a camera module under test using the slanted-edge SFR method of the ISO12233 standard. The Asymmetric Edge Spread Function (AESF) test measures the asymmetry of the Edge Spread Function (ESF) at various field points of a camera module under test using Asymmetry Score (AS).

Spatial Frequency Response (SFR) Setup

Test Hardware Setup

A transmissive chart with D65 light source should be used for SFR testing. The chart and light source must be enclosed in a dark chamber to eliminate all ambient light, glare, and reflections on the chart. The entire test box used for SFR testing should be completely isolated from external vibrations on the factory floor.

Parameter	Value
Illuminant	D65
Light level	Lux level should be adjusted to: 1. >200 cd/m 2. Achieve ±10% uniformity across the entire chart using the 9 measure-ment points shown in Figure 3.
Target	High Contrast Transmissive Apple SFR Target, whose feature pattern is shown in Figure 51. The chart should be fixed on the front side of the glass facing the camera under test. The chart should be the first item in the optical path of the camera, not the glass.
Camera distances to target	Specific distances will be set by the camera module ERS Setup distances must be calibrated accurately with a laser distance meter.
Relay lens	Relay lenses (if any) will be specified in the camera module ERS
Orientation	Specific test orientations will be set by the camera module ERS

Test Chart

The SFR must be measured using the high contrast transmissive Apple SFR chart. This chart is a high contrast chart. An example of the chart pattern can be seen in Figure 51. For detailed chart specifications, please refer to the following document: Apple SFR Test Chart Specifications.

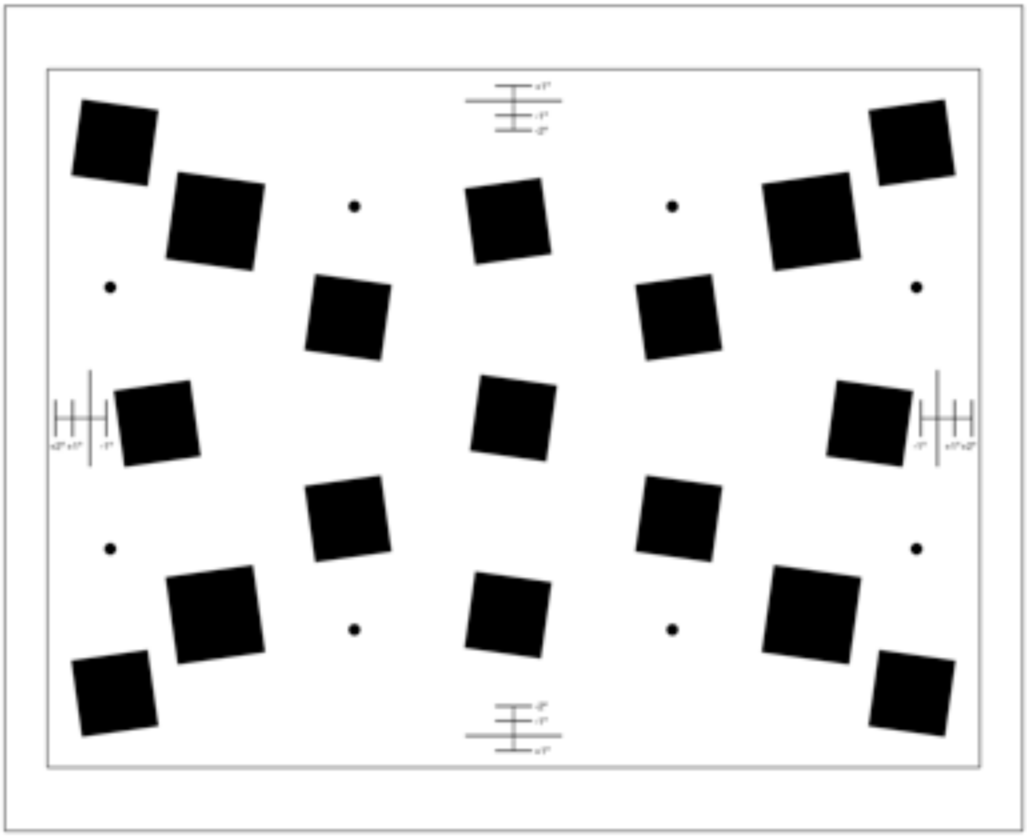


Figure 53. High contrast, transmissive, SFR chart (design pattern subject to change, refer to Apple SFR Test Chart Specifications for further details).

Module Hardware Setup

Module setup should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Value	Comments
Software driven AE	70±5% FSD	average of a 64x64 pixel ROI around the white area which is left of the centre tilted square by 10% of the image width (refer to Figure 1) post pedestal subtraction, on y-channel
Analog gain	1x	
Integration time	<66.67 ms	controlled by AE, limited by frame rate
DPC	off	see ERS for register settings
Frame rate	15 fps	
LSC	off	
AWB	off	

Images of the Apple SFR chart should be captured at the best focus position and not averaged. The best focus position should be determined iteratively by calculating SFR scores for different focus positions and selecting the position that yields the maximum SFR score at the centre (i.e., on-axis) test location, for the highest test frequency (e.g., Ny/4). (Note: While searching for the best focus position, the number of image captures for the SFR measurement can be lowered to one instead of five)

In addition, for the final image to be deemed acceptable, then all off-axis SFR scores must meet the SFR specifications (determined by the module ERS) for all focus positions within ±2VCM_DAC_BITDEPTH / 2APPLE_ISP_AF_BITDEPTH steps from the center peak position (e.g., within ± 4 steps, for VCM_DAC_BITDEPTH = 10 and APPLE_ISP_AF_BITDEPTH = 8).

Alignment

The alignment between the module and the target is critical for the test. Two lasers should be used for calibrating the placement of the chart with respect to the camera module test socket/holder/fixture. The black lines at the edge of the chart frame are to be used for placement of the lasers during calibration. The distance to the chart must be calibrated exactly, using laser distance meters, to the distance under test.

Parameter	Min	Max	Unit
Tilt Alignment (θ)	-0.1	+0.1	Degrees
Rotational Alignment (ϕ)	-1	+1	Degrees

Relay Lens

To keep the production line test machine footprint small, a relay or projective lens can be used. The lens system must be diffraction limited, contain a minimal number of elements, and also be centered/aligned to the centre of the chart and the centre of the camera module test fixture. Relay lens specifications will be provided in the camera module ERS.

Lighting Spectrum

The ideal lighting for the target should provide a continuous spectrum of light that is broad and uniformly distributed across the visible wavelengths. This can be achieved with a light source whose spectral radiance closely matches the standard illuminant D65.

For transmissive charts, an LED light panel may be used, however the spectrum of the LED light source must be measured and approved by Apple. For example, typical white LED panel spectra have a strong peak at blue wavelengths. Thus, in this case, the LED light panel should be specifically constructed to suppress this peak, as shown in Figure 54.

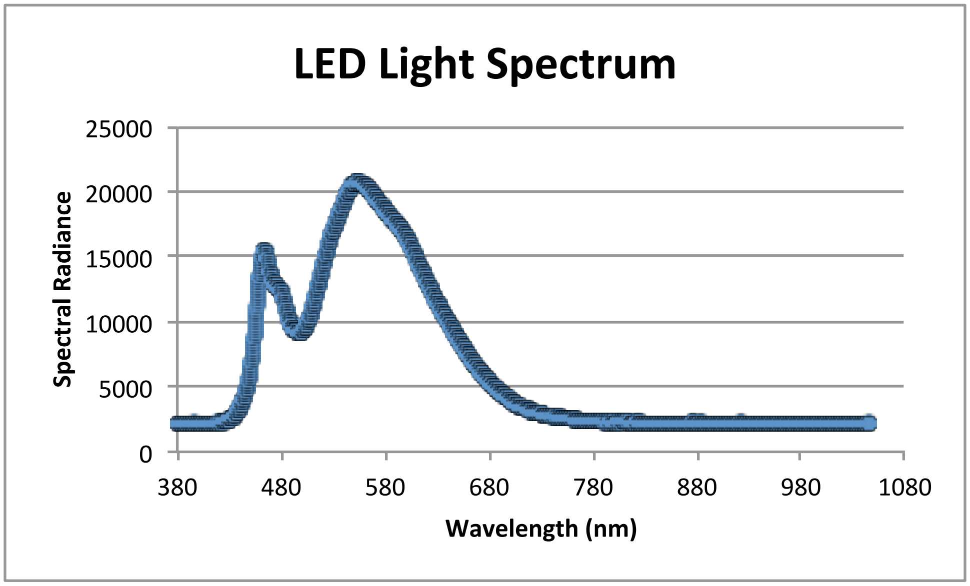


Figure 54. LED light spectrum shape with reduced blue wavelength peak.

Level

The light level necessary to achieve an acceptable camera signal output level depends on the test distance and camera module under test. The luminous emittance should be adjusted to 600 lux at the SFR chart.

Uniformity

The uniformity of the illumination across the active image field must be controlled so that the luminous emittance of the background is within ±10% of the luminous emittance at the centre, using the 10 measurement points shown in Figure 55. The lux measurement in the centre is taken as the average lux of the 4 measurement points near the centre tilted square.

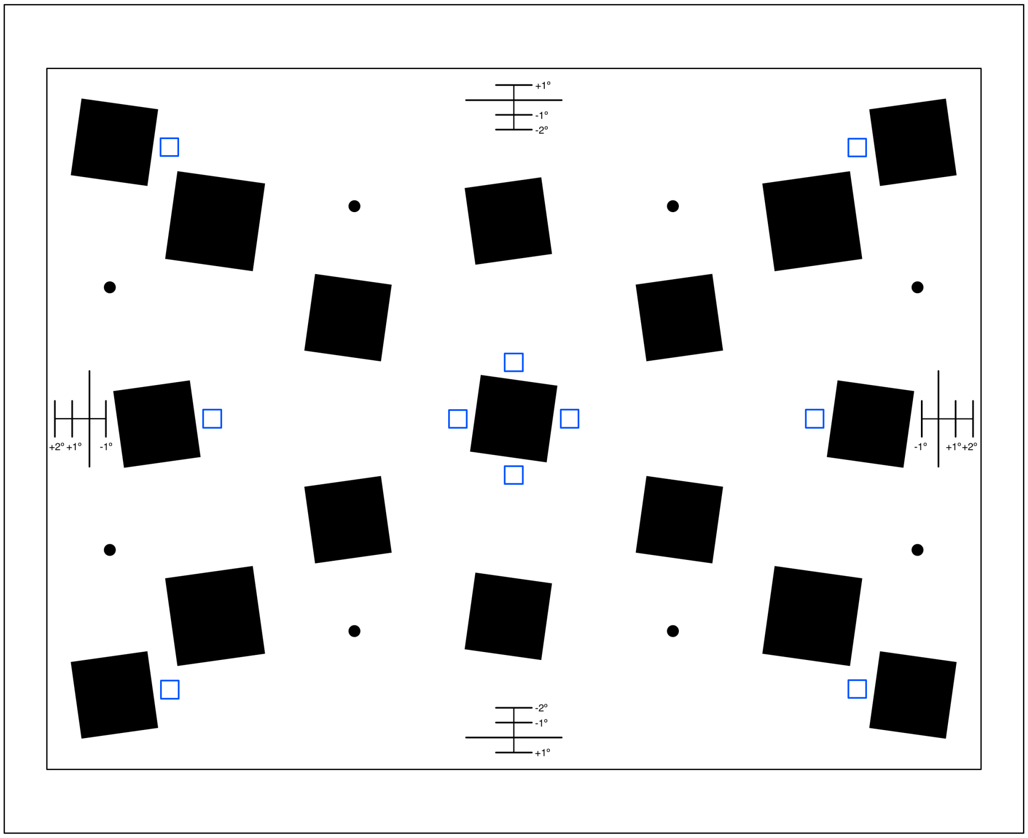


Figure 55: Measurement points for light uniformity check (blue).

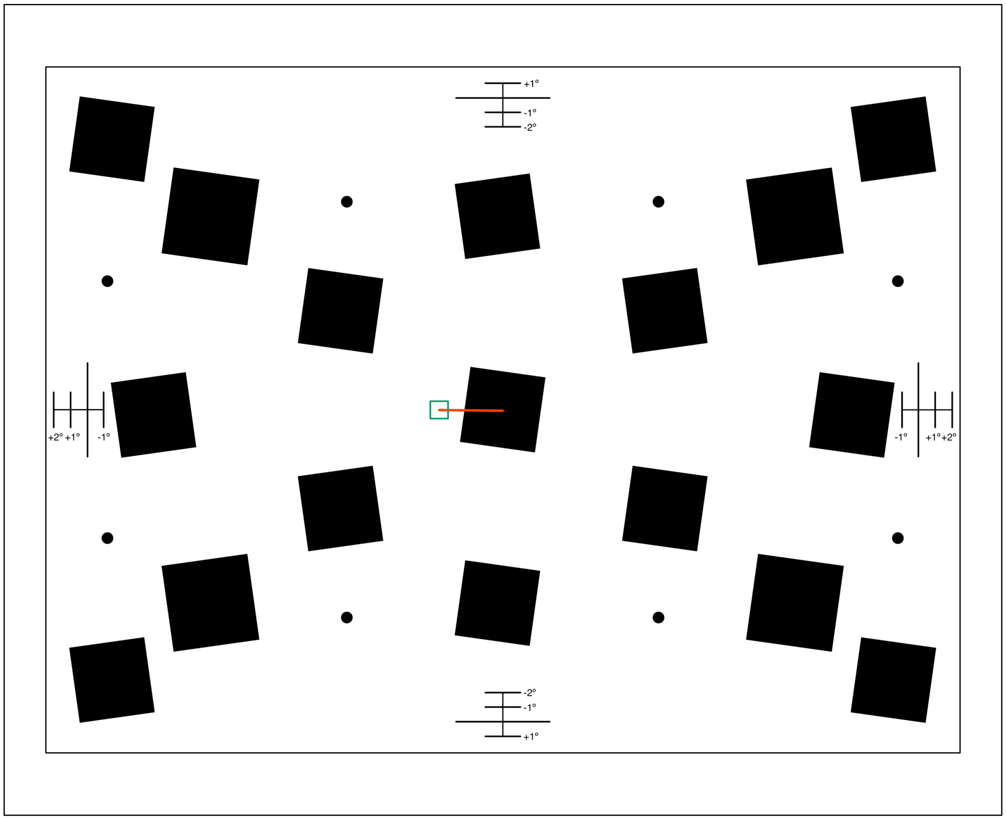


Figure 56. Autoexposure window

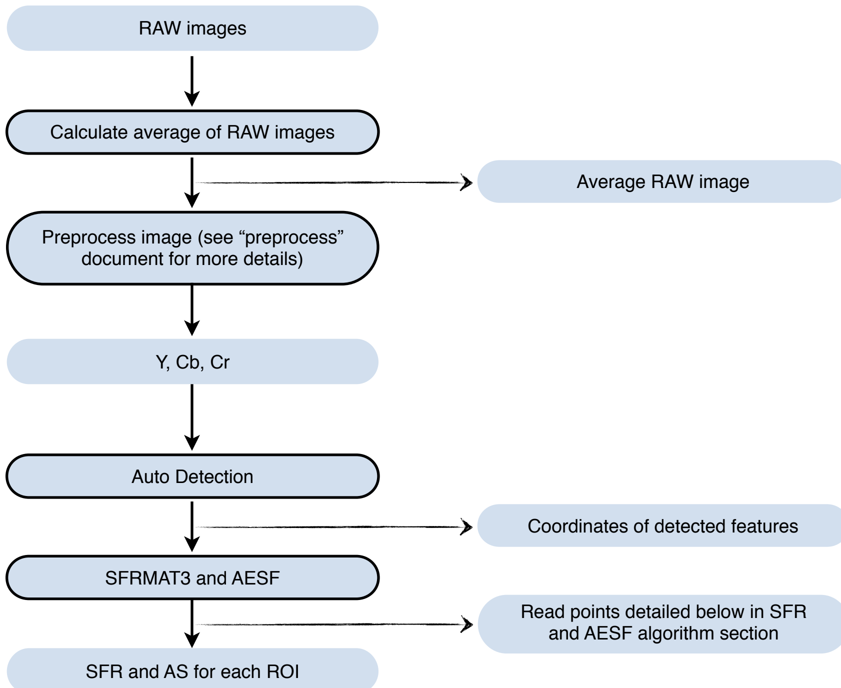
Test Image

The test images should be full scale (e.g., 10-bit) RAW images captured. Multi-frame averaging should be used to reduce temporal noise.

Test Image Specification	Value
Image captures	1 frame
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

For this measurement, the full resolution luminance (Y-channel) image should be used for calculations. All subsequent calculations are to be done with double-precision math. Final output may be rounded to three decimal places.



1. Capture 1 image under the test conditions outlined above.

2. Preprocess the image (see "preprocess" document for more details): subtract pedestal from image, apply white balance & lens shading correction, apply demosaicing, and convert the image from RGB to YCbCr. The output luminance channel (Y) should be used for all subsequent calculations.

3. Perform auto detection of all SFR ROIs, dot features, grayscale patches. Flow chart can be seen in Figure 56.

4. Compute SFR algorithm according to SFRMAT3 (refer to http://losburns.com/imaging/software/SFRedge/sfrmat3_post/index.html).

5. Compute AESF using AESF algorithm.

Auto detection

All the SFR and AESF measurement ROIs and dot features should be automatically extracted. An example algorithm for feature detection is provided in Figure 58.

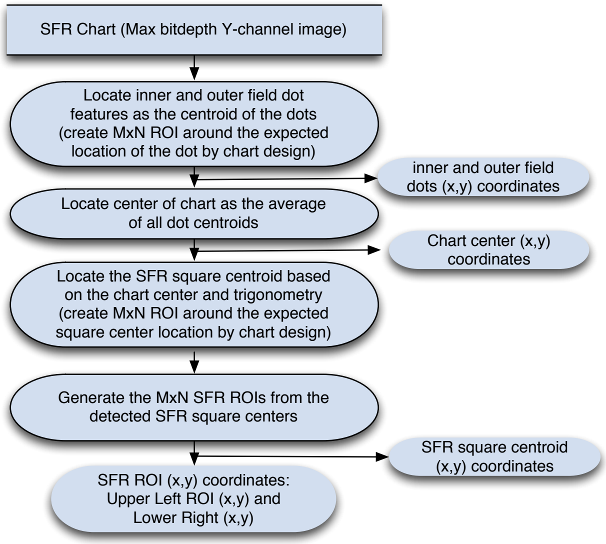


Figure 58. Example algorithm for SFR chart feature auto detection.

1. Detect the chart dot coordinates and compare the dots against the list of ideal chart dot locations.
   These ideal locations are to be generated by using specific diagonal, height, width, angle, and size settings that are provided per program.

2. Generate the ideal chart square center locations in ROI(i) = (x,y) format, where (x,y) is the coordinate pair that describes the square center.
   The list of ideal square locations is based on calculations on the dimensions of the chart and are detailed in getIdealSquares.m

3. Make the ideal square coordinates homogenous by adding an extra column of 1's such that each square center is now described as ROI(i) = (x,y,1)

4. Project the locations of the ideal square centers onto the image by transforming them based on three deltas between the ideal dots and the detected dots. The three transformations each produce a 3x3 matrix. The matrixes are then multiplied in order to produce the final transformation matrix.

   These three calculations are described as follows and can be found in the getAffine.m function:

   4.1. Translation:
   Calculated by differencing the mean of the centers for each set of dots. Value *tx* will be the X component and value *ty* will be the Y component.

$$\text{centerActualDots} = \text{mean(actualDot)}$$
$$\text{centerIdealDots} = \text{mean(idealDot)}$$
$$(\mathrm{t}_x, t_y) = \text{centerActualDots - centerIdealDots}$$

Matrix will be in the form:

$$translation = \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{matrix}$$

4.2. Rotation:

Calculated by calculating the difference of the detected angles from the ideal angles for all 8 dots. Value rotation will be the average of these 8 angle differences.

```
for i=1:8
    rotation_ideal(i) = atand(-(idealDot(i,2)-centerIdealDots(2)) / (idealDot(i,1)-centerIdealDots(1)))
    rotation_actual(i) = atand(-(actualDot(i,2)-centerActualDots(2)) / (actualDot(i,1)-centerActualDots(1)))
end
rotation_ideal(9) = mean(rotation_ideal(1:8))
rotation_actual(9) = mean(rotation_actual(1:8))
rot = (rotation_actual(9) - rotation_ideal(9))
```

Matrix will be in the form:

$$rotation = \begin{matrix} \cos(rotation) & -\sin(rotation) & 0 \\ \sin(rotation) & \cos(rotation) & 0 \\ 0 & 0 & 1 \end{matrix}$$

4.3. Scaling:

Calculated by separating X scaling and Y scaling into two components.

X scaling is calculated by measuring the ratio of the ideal horizontal distance between the dot pairs (1-2), (3-4), (5-6), and (7-8) to the measured distances of those same pairs. Value scaleX will be the average of these four distances.

Y scaling is calculated by measuring the ratio of the ideal vertical distance between the dot pairs (1-4), (2-3), (5-8), and (6-7) to the measured distances of those same pairs. Value scaleY will be the average of these four distances.

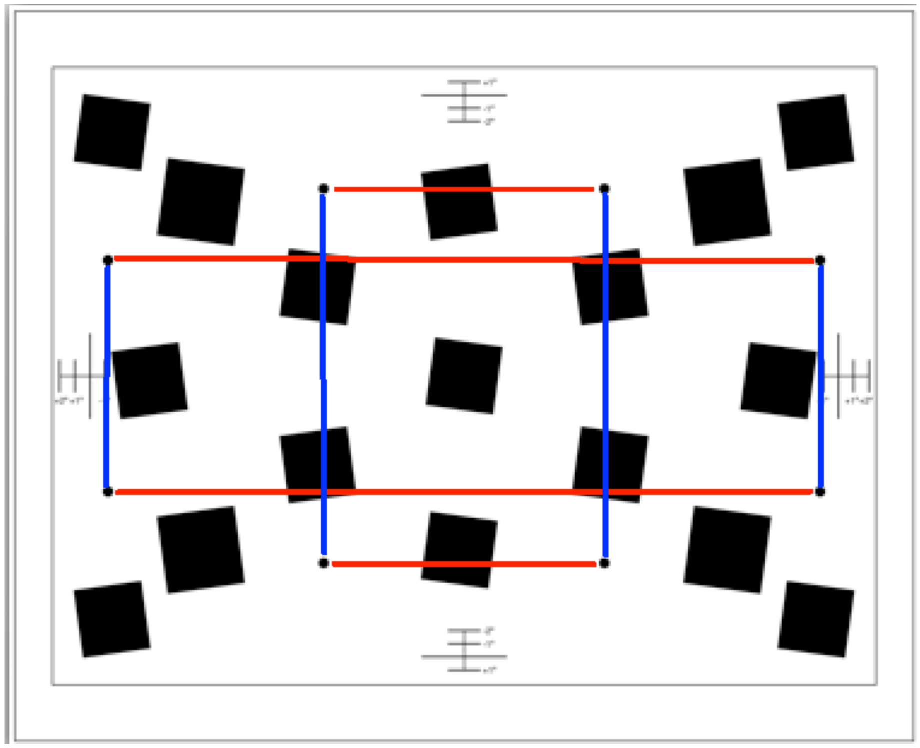


Figure 59: ScaleX (Red) and ScaleY (Blue) distances

Matrix will be in the form:

$$scale = \begin{matrix} scaleX & 0 & 0 \\ 0 & scaleY & 0 \\ 0 & 0 & 1 \end{matrix}$$

4.4. Generate the imageCentering and imageUncentering matrixes by creating two matrixes with the tx and ty compoents set to -(IdealCenter) and (IdealCenter) values. Matrixes should be in the form:

1.

$$imageCentering = \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -idealCenterX & -idealCenterY & 1 \end{matrix}$$

$$imageUncentering = \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ idealCenterX & idealCenterY & 1 \end{matrix}$$

4.5. Multiply the matrices together in the following order to get the final Affine matrix:

$$final = \text{imageCentering} * \text{translateMatrix} * \text{rotationMatrix} * \text{scaleMatrix} * \text{imageUncentering}$$

$$total = \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -idealCenterX & -idealCenterY & 1 \end{matrix} * \begin{matrix} 1 & & 0 \\ & 1 & 0 \\ t_x & t_y & 1 \end{matrix} * \begin{matrix} \cos(rotation) & -\sin(rotation) & 0 \\ \sin(rotation) & \cos(rotation) & 0 \\ 0 & 0 & 1 \end{matrix} * \begin{matrix} scaleX & 0 & 0 \\ 0 & scaleY & 0 \\ 0 & 0 & 1 \end{matrix} * \begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ idealCenterX & idealCenterY & 1 \end{matrix}$$

4.6. Multiply the homogenized idealSquare coordinates as row vectors against this matrix to get the final square center locations.
Eliminate the homogenized column vector so that all square centers are in (X,Y) format.

4.7. Offset the ROIs from the center of each square as detailed in the sfrCreateROITrig.m. Each square can have an ROI placed to the left, right, above, and below by a specific offset.

4.8. Extract the ROIs required by the ERS and perform the SFRMAT3 and AESF calculations.

SFRMAT3 Test Methodology

For each SFR measurement ROI, the SFR values must be computed as per SFRMAT3. Detailed steps of the algorithm are outlined below, and a high level algorithm flow-chart is provided in Figure 60.

Figure 60. Overview of SFRMAT3 and AESF Algorithm for one MxN Y-channel auto detected slanted edge ROI.

NOTE: *The intermediate outputs of the following algorithm must be checked against the results from Apple's MATLAB adaptation of SFRMAT3 for 2 ROIs (one with vertical edge, one with horizontal edge).*

5. Check if Edge Spread Function (ESF) image needs to be rotated based on the edge orientation in the ROI to be processed. If the edge is horizontal, the ESF image must be rotated by 90° to make the edge vertical prior to subsequent processing.

6. Estimate edge slope and offset from ESF image via edge detection

6.1. Generate Line Spread Function (LSF) image by differentiating each row of the ESF image via application of a Finite Impulse Response (FIR), "first difference" filter [0.5, -0.5]

6.2. Roughly detect edge slope and offset by fitting a line through the centroids of each row of the LSF image.

6.2.1. For each row of the LSF image, apply a Hamming window whose centre is the centre of that row. Compute the centroid of the windowed row, compensating the coordinate for the displacement effect of the FIR filter phase (i.e., subtract -0.5 pixels from the detected centroid coordinate). Perform a linear fit to the centroid coordinates across each row to determine the slope and offset.

6.3. Fine detection of edge slope and offset by fitting another line through the centroids of each row of the LSF image.

6.3.1. For each row of the LSF image, apply a Hamming window whose centre is the estimated centroid for the given row (which can be computed from the rough detection linear fit). Compute the centroid of the windowed row. Perform a linear fit to the centroid coordinates across each row to determine a refined slope and offset value for the edge.

7. ESF image row truncation

   7.1. Truncate the number of rows in the ESF image so that the number of cycle transitions of the linear fit to the edge is an integer number (i.e. the linear fit must end on a pixel (i.e. ending on sub-pixels is not allowed)

8. Generate the super-sampled ESF via projection operation

   8.1. Project the ESF image data to the line which is both normal to the detected edge and passing through it's midpoint, binning pixels into one of 4 bins depending on their distance to the edge. After all pixels have been projected to the bins, average the bins to generate the super-sampled ESF.

9. Generate the super-sampled LSF via differentiation

   9.1. Apply 3 tap FIR filter kernel [-0.5, 0.0, 0.5] to the super-sampled ESF to generate the super-sampled LSF

   9.2. Centre the super-sampled LSF so it is in the centre of it's own data array by determining the centroid of the super-sampled LSF and shifting the data appropriately

10. Apply frequency domain transformation to a Hamming windowed, super-sampled LSF. A Hamming window whose centre is the centre of the oversampled LSF (4 x ROI_WIDTH + 1)/2 data array should be generated and applied to the super-sampled LSF. Apply one dimensional Discrete Fourier Transform (DFT) to the windowed, super-sampled LSF.

11. Compute SFR value by normalizing the magnitude of the DFT output by the area under the super-sampled LSF curve and multiplying by a scale factor which compensates for the use of the 3-tap FIR filter used to generate the super-sampled LSF.

12. Generate the spatial frequency array associated with each entry of the computed DFT array as f(k) = k/(Nd), for k = 0 to N-1. Compensate the spatial frequency sampling interval for sampling normal to the detected edge by multiplying the frequencies by the constant d = cos(atan(slope)), where slope is the edge slope from the fine detection of Step 2.

13. Determine SFR at the target test spatial frequencies. Due to discrete sampling of the spatial frequency axis, linear interpolation between the 2 nearest frequencies to the target test frequency is used to determine the SFR at specific test frequencies of interest (i.e. Ny, Ny/2, Ny/4, Ny/8).

14. Smooth super-sampled ESF from step 8 using 1x5 average filter [1/5 1/5 1/5 1/5 1/5] to reduce noise.

15. Apply 3 tap FIR filter kernel [-0.5, 0.0, 0.5] to the smoothed super-sampled ESF to generate the super-sampled LSF.

16. Compute maximum super-sampled LSF value pixel location and assign as split location for separating left and right side of ESF.

17. Mirror ESF from step 8 about ESF(SPLIT_LOC), then mirror the mirrored ESF about SPLIT_LOC. Adjust for any offsets and missing data from the mirroring operations by shifting and extending data appropriately. The ESF size and ESF(SPLIT_LOC) value should be preserved after the mirroring operations.

18. Generate 2 ESFs by concatenating left and right side of the original ESF with the right and left side of the mirrored ESF, respectively.

19. Repeat steps 9 to 12 for each of the 2 ESFs to obtain their spatial frequency arrays.

20. Report AS (Asymmetry Score) for AESF by calculating the SFR difference between the 2 ESFs at select test spatial frequencies of interest, where the minuend is the SFR of the edge side closest to the SFR chart image center. The average AS for a range of sampled frequencies can be calculated by adding the SFR differences at the frequencies and dividing by the number of frequencies sampled. Due to discrete sampling of the spatial frequency axis, linear interpolation between the 2 nearest frequencies to the target test frequency is used to to determine the SFR at specific test frequencies (i.e. Ny, Ny/2, Ny/4, Ny/8).

Algorithm Parameter Settings

The ROIs for SFR measurements should be tuned per camera program. The optimal ROI shape and size is a tradeoff between balancing the effects of noise with the effects of line spread function truncation. To limit the effect of noise influencing the measurement, it is best to decrease the width of the ROI to an area around the edge transition. Decreasing the width too much though has the detrimental effect of artificially boosting the measured SFR value over all spatial frequencies due to the normalization step in the SFR algorithm. Thus, the ROI can be tuned empirically to balance these two effects. A specific M x N ROI for SFR testing will be specified in the camera module ERS.

Algorithm ROI Locations

The ROI's for the SFR measurements are dependent upon the program's SFR chart. The ROI's should be centered on the middle of the edge of each SFR edge. When the MxN ROI is not equal, the longer side of the ROI should lie parallel to the edge such that the maximum edge is contained in the ROI.

Once the fiducial dots of every image are detected, the ROI's center position shall be calculated as per the documentation provided in the Chart Specifications. Figure 61 shows an example of the ideal ROI's as generated by the flowchart in Figure 58.

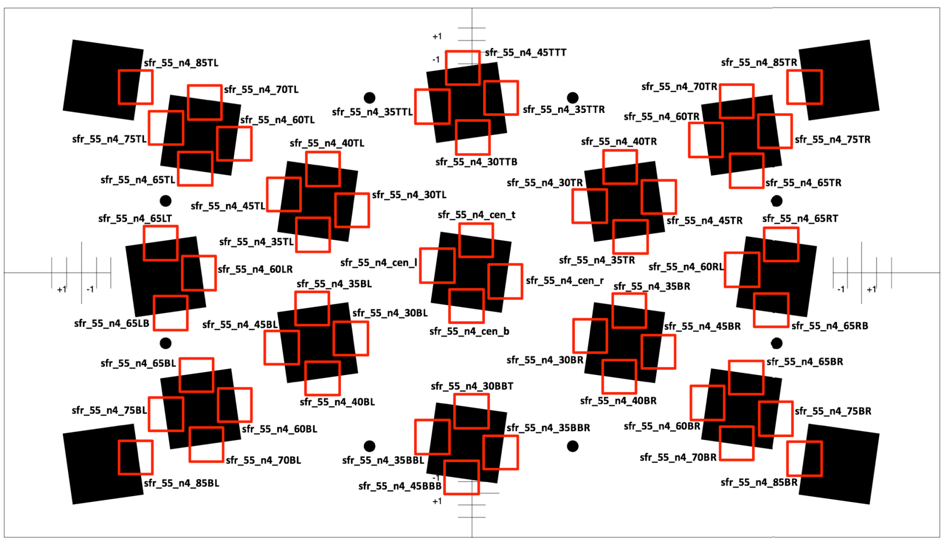


Figure 61. SFR ROI measurement locations

Results Naming Convention in Log File

The SFR and AESF test results for an image must be reported in a CSV file with the following header symbols. A description of what each symbol refers to is given in Tables 3 and 4. An ROI map is also provided in Figure 61.

Table 3. SFR measurement ROI CSV header symbols

X denotes test distance in cm (i.e. replace “X” with “30” or “60”), Z denotes all SFR ROI locations (i.e. replace with 60TL, 65TL, etc.)

Symbol	Description	Symbol	Description
sfr_X_n8_60TL	60% TL right @ Ny/8	sfr_X_n4_60TL	60% TL right @ Ny/4
sfr_X_n8_65TL	60% TL bottom @ Ny/8	sfr_X_n4_65TL	60% TL bottom @ Ny/4
sfr_X_n8_70TL	60% TL top @ Ny/8	sfr_X_n4_70TL	60% TL top @ Ny/4
sfr_X_n8_75TL	60% TL left @ Ny/8	sfr_X_n4_75TL	60% TL left @ Ny/4
sfr_X_n8_60TR	60% TR left @ Ny/8	sfr_X_n4_60TR	60% TR left @ Ny/4
sfr_X_n8_65TR	60% TR bottom @ Ny/8	sfr_X_n4_65TR	60% TR bottom @ Ny/4
sfr_X_n8_70TR	60% TR top @ Ny/8	sfr_X_n4_70TR	60% TR top @ Ny/4
sfr_X_n8_75TR	60% TR right @ Ny/8	sfr_X_n4_75TR	60% TR right @ Ny/4
sfr_X_n8_cen_l	0% center left @ Ny/8	sfr_X_n4_cen_l	0% center left @ Ny/4
sfr_X_n8_cen_r	0% center right @ Ny/8	sfr_X_n4_cen_r	0% center right @ Ny/4
sfr_X_n8_cen_t	0% center top @ Ny/8	sfr_X_n4_cen_t	0% center top @ Ny/4
sfr_X_n8_cen_b	0% center bottom @ Ny/8	sfr_X_n4_cen_b	0% center bottom @ Ny/4
sfr_X_n8_cen	0% average of center regions @ Ny/8	sfr_X_n4_cen	0% average of center regions @ Ny/4
sfr_X_n8_60BL	60% BL right @ Ny/8	sfr_X_n4_60BL	60% BL right @ Ny/4
sfr_X_n8_65BL	60% BL top @ Ny/8	sfr_X_n4_65BL	60% BL top @ Ny/4
sfr_X_n8_70BL	60% BL bottom @ Ny/8	sfr_X_n4_70BL	60% BL bottom @ Ny/4
sfr_X_n8_75BL	60% BL left @ Ny/8	sfr_X_n4_75BL	60% BL left @ Ny/4
sfr_X_n8_60BL	60% BR right @ Ny/8	sfr_X_n4_60BL	60% BR right @ Ny/4
sfr_X_n8_65BL	60% BR top @ Ny/8	sfr_X_n4_65BL	60% BR top @ Ny/4

Symbol	Description	Symbol	Description
sfr_X_n8_70BL	60% BR bottom @ Ny/8	sfr_X_n4_70BL	60% BR bottom @ Ny/4
sfr_X_n8_75BL	60% BR left @ Ny/8	sfr_X_n4_75BL	60% BR left @ Ny/4
sfr_X_n8_85TL	85% TL @ Ny/8	sfr_X_n4_85TL	85% TL @ Ny/4
sfr_X_n8_85TR	85% TR @ Ny/8	sfr_X_n4_85TR	85% TR @ Ny/4
sfr_X_n8_85BL	85% BL @ Ny/8	sfr_X_n4_85BL	85% BL @ Ny/4
sfr_X_n8_85BR	85% BR @ Ny/8	sfr_X_n4_85BR	85% BR @ Ny/4
sfr_X_n8_30TL	30% TL @ Ny/8	sfr_X_n4_30TL	30% TL @ Ny/4
sfr_X_n8_30TR	30% TR @ Ny/8	sfr_X_n4_30TR	30% TR @ Ny/4
sfr_X_n8_30BL	30% BL @ Ny/8	sfr_X_n4_30BL	30% BL @ Ny/4
sfr_X_n8_30BR	30% BR @ Ny/8	sfr_X_n4_30BR	30% BR @ Ny/4
sfr_X_n8_60L	60% L @ Ny/8	sfr_X_n4_60L	60% L @ Ny/4
sfr_X_n8_60R	60% R @ Ny/8	sfr_X_n4_60R	60% R @ Ny/4
sfr_X_n8_30T	30% T @ Ny/8	sfr_X_n4_30T	30% T @ Ny/4
sfr_X_n8_30B	30% B @ Ny/8	sfr_X_n4_30B	30% B @ Ny/4
sfr_X_n8_30_TBdelta	Average of 30% TL&TR minus average of 30% BL&BR for ny/8	sfr_X_n4_30_TBdelta	Average of 30% TL&TR minus average of 30% BL&BR for ny/4
sfr_X_n8_30_LRdelta	Average of 30% TR&BR minus average of 30% TL&BL for ny/8	sfr_X_n4_30_LRdelta	Average of 30% TR&BR minus average of 30% TL&BL for ny/4
sfr_X_n8_30_delta	Max minus Min of all 30% field points	sfr_X_n4_30_delta	Max minus Min of all 30% field points for ny/4
sfr_X_n8_60_TBdelta	Average of 60% TL&TR minus average of 60% BL&BR for ny/8	sfr_X_n4_60_TBdelta	Average of 60% TL&TR minus average of 60% BL&BR for ny/4
sfr_X_n8_60_LRdelta	Average of 60% TR&BR minus average of 60% TL&BL for ny/8	sfr_X_n4_60_LRdelta	Average of 60% TR&BR minus average of 60% TL&BL for ny/4
sfr_X_n8_60_delta	Max minus Min of all 60% field points for ny/8	sfr_X_n4_60_delta	Max minus Min of all 60% field points for ny/4

Symbol	Description	Symbol	Description
sfr_X_n8_85_TBdelta	Average of 85% TL&TR minus average of 85% BL&BR for ny/8	sfr_X_n4_85_TBdelta	Average of 85% TL&TR minus average of 85% BL&BR for ny/4
sfr_X_n8_85_LRdelta	Average of 85% TR&BR minus average of 85% TL&BL for ny/8	sfr_X_n4_85_LRdelta	Average of 85% TR&BR minus average of 85% TL&BL for ny/4
sfr_X_n8_85_delta	Max minus Min of all 85% field points for ny/8	sfr_X_n4_85_delta	Max minus Min of all 85% field points for ny/4
as_X_n4_Z	Z @ Ny/4		

Temporal Noise

This document provides a scalable metric for characterizing the temporal noise of the sensor over a given sequence of images.

Test Hardware Setup

A D50 light source fitted with an opal diffuser (per Appendix A) and a dark field should be used as the test targets.

Module Hardware Setup

Module set up should follow module ERS default settings. In addition, the settings below supersede any default settings:

Parameter	Light Field	Dark	Comments
Lens position	focused @ 2m	-	Lens position must be in a deterministic position for D50.
software driven AE	80%±5% FSD	Off	Use average of central 100x100 green pixels (an average of Gr and Gb pre-pedestal subtraction) for SW exposure adjust.
analog gain	1x	Max gain	Maximum sensor gain (e.g., 16x) should be used
integration time	<66 ms	<1ms	Light field capture integration time set to meet AE LSB target, dark field capture set to the minimum integration time
DPC	on	on	Turn DPC on according to ERS
frame rate	15 fps	30fps	
LSC	disabled	disabled	
AWB	disabled	disabled	

Test Images

The test images should be full scale (e.g., 10-bit) RAW images captured with module LSC and AWB turned off.

Test Image Specification	Value
Image captures	Five 1 frame captures.
Image bit-depth	Full scale bit-depth (e.g., 10-bit)
Image format	RAW

Test Methodology

For this measurement the calculations should be performed on the RAW Bayer image, colour channel separation is not necessary. All subsequent calculations are to be done with double-precision math. Final output may be rounded to three decimal places.

1. Capture 5 RAW images, *raw1*, *raw2, ... raw5*. Subtract the pedestal and perform no other preprocessing steps.
2. Calculate the averaged frame of the 5 images and the mean value of the averaged frame and call these values:

   averageFrame, *frameSetMean*

3. From the averageFrame, calculate the average for each row and column, and call these values:

   rowFrameMean, colFrameMean,

4. From the averageFrame and each individual frame in the set, calculate the total sum of the squared noise and the total sum of the squared temporal noise according to the following calculations:

   noiseTotalSquArray = sum((IDFrameSet - frameSetMean).^2,3);

   noiseTotalTempSquArray = sum((IDFrameSet -averageFrame).^2,3);

   noiseTotalSqu = sum(sum(*noiseTotalSquArray*,2));

   noiseTotalTempSqu = sum(sum(*noiseTotalTempSquArray*,2));

5. Calculate the total noise and total temporal noise as the square root of the normalized sums from the previous step as follows:

   totalNoise = sqrt(*noiseTotalSqu*/(nFrames * h * w));

   totalNoiseTemp = sqrt(*noiseTotalTempSqu*/(nFrames * h * w));

6. From each individual RAW frame in the set, calculate the row means and call it:

   rowFrameMeanSet

7. Calculate the sum of the total row squared noise and the total sum of the row temporal noise according to the following calculations:

   noiseRowTotalSquArray = sum((*rowFrameMeanSet* - *frameSetMean*).^2,3);

   noiseRowTempSquArray = sum((*rowFrameMeanSet*(:,1,:)-*rowFrameMean*).^2,3);

   noiseRowTotalSqu = sum(*noiseRowTotalSquArray*);

   noiseRowTempSqu = sum(*noiseRowTempSquArray*);

8. Calculate the total row noise and total row temporal noise as the square root of the normalized sums from the previous step as follows:

rowTotal = sqrt(*noiseRowTotalSqu*/(h * nFrames));

rowTemp = sqrt (*noiseRowTempSqu*/(h * nFrames));

9. From each individual frame in the set, calculate the column means and call it:

colFrameMeanSet

10. Calculate the sum of the total row squared noise and the total sum of the row temporal noise according to the following calculations:

noiseColTotalSquArray = sum((*ColFrameMeanSet* - *frameSetMean*).^2,3);

noiseColTempSquArray = sum((*ColFrameMeanSet*(:,1,:)-*colFrameMean*).^2,3);

noiseColTotalSqu = sum(*noiseColTotalSquArray*);

noiseColTempSqu = sum(*noiseColTempSquArray*);

11. Calculate the total column noise and total column temporal noise as the square root of the normalized sums from the previous step as follows:

colTotal = sqrt(*noiseColTotalSqu*/(h * nFrames));

colTemp = sqrt (*noiseColTempSqu*/(h * nFrames));

12. Calculate the pixel temporal noise as the remainder of the total temporal noise minus the row temporal noise and column temporal noise as follows:

pixelTemp = sqrt(abs(*totalNoiseTemp*^2- *rowTemp*^2 - *colTemp*^2));

13. Calculate the temporal row noise and column noise ratio as the ratio between the pixel temporal noise and the row noise and column noise as follows:

tempRowNoiseRatio = *pixelTemp*/*rowTemp*;

tempColNoiseRatio = *pixelTemp*/*colTemp*;

Test Methodology Flowchart

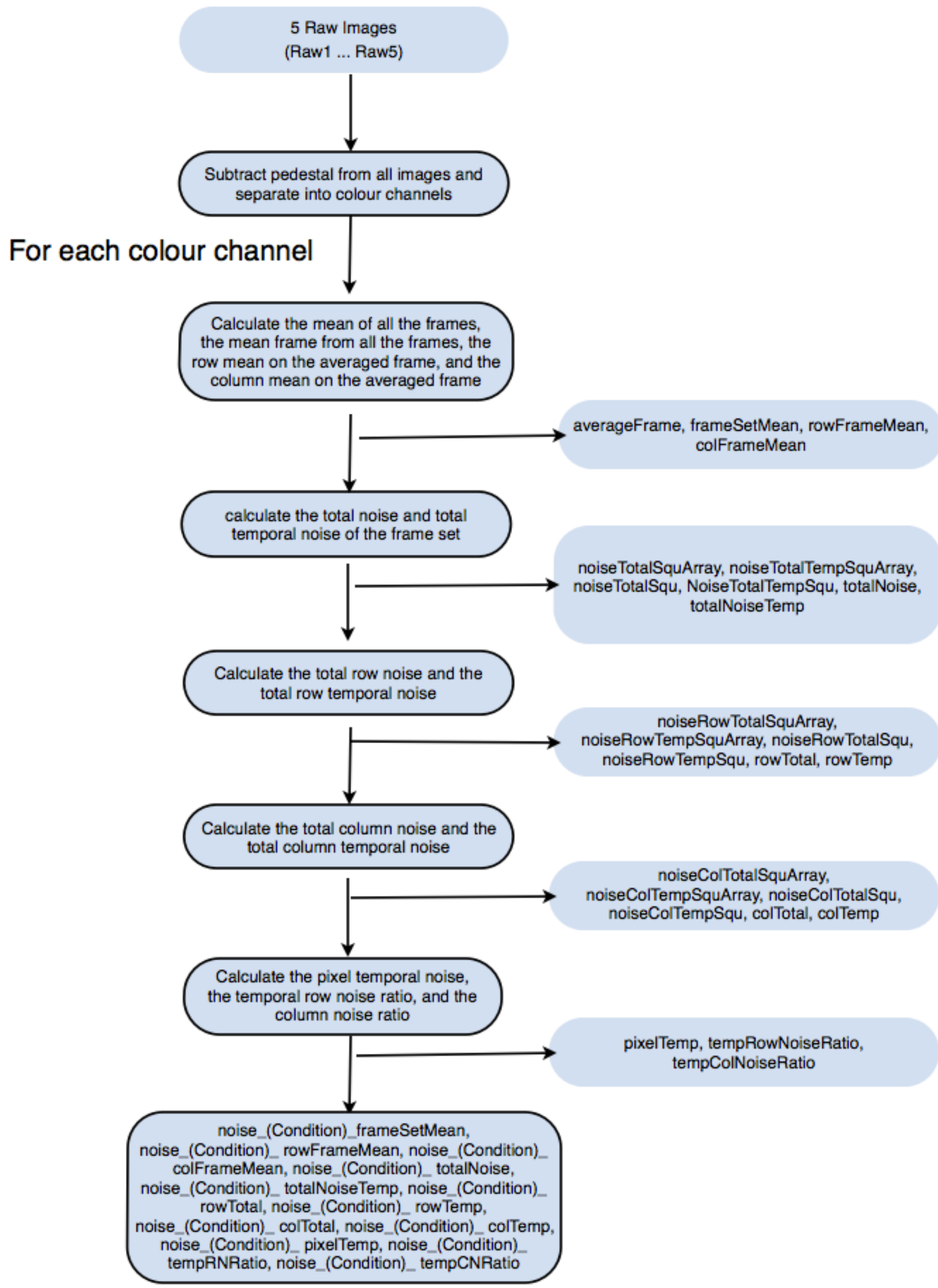


Figure 60 : Temporal Noise Methodology Flow Chart

Results Naming Convention in Log File

Name	Description	Unit
noise_(Condition)_frameSetMean_*	The mean of all the pixels in the averaged frame.	LSB
noise_(Condition)_ rowFrameMean_*	The mean of all the rows in the averaged frame.	LSB
noise_(Condition)_ colFrameMean_*	The mean of all the columns in the averaged frame.	LSB
noise_(Condition)_ totalNoise_*	The total noise in the frame set.	LSB
noise_(Condition)_ totalNoiseTemp_*	The total temporal noise in the frame set.	LSB
noise_(Condition)_ rowTotal_*	The total row noise in the frame set.	LSB
noise_(Condition)_ rowTemp_*	The total row temporal noise in the frame set.	LSB
noise_(Condition)_ colTotal_*	The total column noise in the frame set.	LSB
noise_(Condition)_ colTemp_*	The total column temporal noise in the frame set.	LSB
noise_(Condition)_ pixelTemp_*	The total pixel temporal noise in the frame set.	LSB
noise_(Condition)_ tempRNRatio_*	The ratio of pixel temporal noise to row temporal noise in the frame set.	LSB
noise_(Condition)_ tempCNRatio_*	The ratio of pixel temporal noise to column temporal noise in the frame set.	LSB

_* = the colour channel of the measurement

(Condition) = the light condition (dark / D50) of the image capture

VCM Actuator Testing

Actuator testing provides a set of metrics used to characterize the actuator. This data is critical to ensuring the correct operation of this component for autofocus. Actuator issues may lead to inability to focus.

Actuator Setup

Test Hardware Setup

Test setup is designed to ensure accurate and robust results. Careful selection of the laser distance meter ensures a robust production implementation.

Parameter	Value
Target	Top surface of lens barrel
Distance to target	Optimized to ensure use of full laser meter range
Orientation	Module up

Parameter for Laser Meter	Value
Minimum resolution	0.1µm
Measurement range	0-500µm (minimum range required)
Sampling period	0.33 ms
Averaging	64 samples/read point
Additional features required	zero-ing function

Module Hardware Setup

Module control should be done directly through the actuator driver.

Parameter	Value	Comments
Step size	<=1mA	Value needs to be as close to 1mA as possible without exceeding 1mA.
Start step	0 mA	
End step	max driver current	Maximum current, as defined by the actuator driver

Test Methodology

1. Apply 0mA to the actuator driver and take a reading with the laser distance meter (for trouble-shooting purposes only).

    vcm_initial_disp = VCM displacement (in μm) at 0mA

2. Zero the laser distance meter.
3. Increase the current in increments of the step size up to end step and then decrement the current back down to start step, using the same step size. The displacement of the lens barrel from the zero position should be recorded at each step and the output data should be recorded in μm. Store the values using the following variable names:

vcm_up	VCM displacement values in the positive direction (this is a vector)
vcm_down	VCM displacement values in the negative direction (this is a vector)

4. Define the following parameters for the up (positive start step) direction, with xx denoting the particular current or displacement value listed in the ERS, using the appropriate unit for the value:

vcm_rangeLow_xxmA	closest measurement point (in mA) below the lowest current (xx mA) specified in the ERS
vcm_rangeHigh_yymA	closest measurement point (in mA) above the highest current (yy mA) specified in the ERS
vcm_rangeLow_xxμm	closest measurement point (in μm) below the lowest displacement (xx μm) specified in the ERS
vcm_rangeHigh_yyμm	closest measurement point (in μm) above the highest displacement (yy μm) specified in the ERS

For each range value, a corresponding current (or displacement) measurement applies:

vcm_xxlow_μm	actual displacement measurement (in μm) corresponding to the VCM current stored in the parameter vcm_rangeLow_xxmA
vcm_yyhigh_μm	actual displacement measurement (in μm) corresponding to the VCM current stored in the parameter vcm_rangeHigh_yymA
vcm_xxlow_mA	actual current measurement (in mA) corresponding to the displacement stored in the parameter vcm_rangeLow_xxum
vcm_yyhigh_mA	actual current measurement (in mA) corresponding to the displacement stored in the parameter vcm_rangeHigh_yyμm

5. Calculate the follow parameters listed in the Log File Output table:
   (Note: These should all be calculated in the positive step direction unless otherwise noted)

Results Naming Convention in Log File

* Note: curly brackets { } are used to denote variables that are vectors, and not single scalar values.

Name	Description	unit
vcm_lowstroke_um	The average stroke value in {vcm_up, vcm_down}, considering only the stroke values for current values between 0 mA and the value stored in the parameter vcm_rangeHigh_xxmA (where xx is the maximum current specified by the ERS).	µm
vcm_xxstrokeMax_um	The average total stroke value in {vcm_up, vcm_down} considering the entire current range (i.e. 0 mA to xx mA, where xx is the maximum current defined by the VCM driver).	µm
vcm_sensitivity_* Where * is up / down.	The actual estimated slope of displacement vs. input current for the linear region of operation for the VCM. The slope is calculated by taking the first order derivative of the vcmUp and vcmDown data sets and smoothing it with a filter of [0.25 0.5 0.25]. The points where the derivative cross above and then below the ideal_stepSize have BackstepSize added to the idxLow and subtracted from idxHigh to define the linear region. The linear region is then used to calculate a slope using a least squares estimate of the data between the two indexes. Ideal_stepSize is the minimum allowed value of the vcm sensitivity. BackStepSize is the VCM specific DAC steps needed to re-enter the linear region. The slope of the least squares estimate denotes the sensitivity of the vcm and must be calculated for both up and down directions.	um/mA

Name	Description	unit
vcm_startCurrent_mA	The idealized current required to begin displacement of the VCM. This value is calculated by extending the linear region of the VCM's sensitivity calculation and projecting to the theoretical intersection point with the x-axis. where b is the y-intercept of the line (vcm_ideal) used to model the linear region of motion for the VCM. (Note: b will most likely be a negative value.) The start current is only calculated based on the vcm_sensitivity_up measurement.	um/mA
vcm_startStep_µm	The actual physical displacement of the VCM corresponding to vcm_startCurrent_mA (see previous definition for vcm_s-tartCurrent_mA)	µm

Name	Description	unit
vcm_jump_btm_µm	The maximum single-step displacement of the VCM (in either the positive or the negative direction) in the lower half of the full displacement range. (This value should be signed, to denote if the jump step happened during the up-stroke (positive) or down-stroke (negative) direction.) Jumpstep_RangeLow = [RangeLow_Low, RangeLow_High] Where RangeLow_Low and RangeLow_High are parameters detailed in the ERS	µm
vcm_jump_btm_DAC	The starting DAC value associated with vcm_jump_btm_µm	DAC

Name	Description	unit
vcm_jump_top_µm	The maximum single-step displacement of the VCM (in either the positive or the negative direction) in the upper half of the full displacement range. (This value should be signed, to denote if the jump step happened during the up-stroke (positive) or down-stroke (negative) direction.) Jumpstep_RangeHigh = [RangeHigh_Low, RangeHigh_High] Where RangeHigh_Low is a parameters described in the ERS and RangeHigh_High is equal to the value of vcm_Stroke_lowmA	µm
vcm_jump_top_DAC	The starting DAC value associated with vcm_jump_top_µm	DAC
vcm_linearity_*_µm Where * is up/down directions	The maximum difference between the measured displacement curve from each {vcm_up, vcm_down} and the ideal displacement curve for the same current setting calculated using the vcm_sensitivity_* for the appropriate direction. The linearity is only calculated within the linear region of the VCM's displacement-vs-current characteristic curve, i.e. for vcm_rangeLow_xxmA <= current <= vcm_rangeHigh_yymA -where {vcm_up, vcm_down} - {vcm_ideal_up, vcm_ideal_down} denotes element-by-element subtraction of the two vectors.	µm
vcm_hysteresis_µm	The maximum difference in the VCM's physical position when moving in the positive direction vs. moving in the negative direction. The sign of this number represents the direction of the vcm hysteresis, IE: positive numbers denote hysteresis during to movement up and negative numbers denote hysteresis during the movement down. for vcm_rangeLow_xxmA <= current <= vcm_rangeHigh_yymA	µm

Name	Description	unit
vcm_hysteresis_DAC	DAC value corresponding to the current level associated with vcm_hysteresis_µm	DAC

Appendix A: Flat Field Light Source

For each colour calibration station, two light sources are required: one source for “A Light” 2800K and one for “D50” 5000K colour temperatures. Each light source will be connected to a PC running camera module calibration software.

System Specifications

Parameter	Light Source 1	Light Source 2
Light Source x,y range after calibration	CCT = 2800K +- 20K x = 0.457 +/- 0.019 y = 0.419 +/- 0.016	CCT = 5000K +- 20K x = 0.347 +/- 0.019 y = 0.378 +/- 0.019
IR Cut Filter	none in the system	none in the system
Bulb and Replacements	Halogen, see Material Suppliers	Halogen, see Material Suppliers
D50 Filter	none in the system	HOYA LB-120, see Material Suppliers
Camera Distance From Light Source	8mm - 16mm	8mm - 16mm
Camera x,y From Centre	+- 5mm	+- 5mm
Uncovered Area at Diffuser Surface	65mm x 65mm	65mm x 65mm
Connection to Calibration PC	DAQ, see Calibration Equipment List	DAQ, see Calibration Equipment List

Expected Characteristics

Parameter	Light Source 1	Light Source 2
Light Source Spatial Non-Uniformity in central 25mm	Luminance variation: < 3% Colour: 2800K +/- 10K	Luminance variation: < 3% Colour: 5000K +/- 10K
Light DC Regulation	< +/- 0.5% p-p from full scale	< +/- 0.5% p-p from full scale
Luminance range (i1Pro2 measured)	300-1200 cd/m	100-300 cd/m

Material Suppliers

	Part Number	Manufacturer	Description
1	DC-950HB	Dolan-Jenner http://www.dolan-jenner.com/PDFS/DC950H_spec.pdf	Fiber optic light source, 14.99mm ferrule interface, no iris, no IR filter installed
2	EKE-XPB 0100905-800 01	Dolan-Jenner custom part for Apple integrators	Pre-burned long-life halogen bulb, to be replaced every 2 months
3	QVABL48M	Dolan-Jenner http://www.dolan-jenner.com/Pro/Backlights.htm	Fiber optic backlight, 14.94mm end ferrule, 1219.2mm cable length, 85.6mm x 108mm window
4	68600902770	Dolan-Jenner	Quick disconnect lamp socket, to be replaced every 2 years (first one included in DC-950HB)
5	LB-120	HOYA Optics http://www.hoyaoptics.com/color_filter/light_balancing.htm Contact: Bruce Au Sales Manager Hoya Corporation USA - Optics Division 3285 Scott Blvd Santa Clara, CA 95054 Tel: 408-654-2272 Fax:408-654-2270	A to D50 light balancing filter, custom size 85mm x 85mm x 3mm melt 1-8416 for production, allow 4-5 week lead time (pre-existing melt 1-8270 is acceptable) short-term lead time is available, using different melt Note: only needed for D50 light boxes, 2800K A light boxes don't have this filter installed
6	3" x 3" mirrors, item #163388	Factory Direct Craft http://factorydirectcraft.com	76.2mm x 76.2mm x 1.5mm mirrors, quantity 4 needed for walls of light box
7	0075-0075-0060-GV-CA 0075-0075-0030-GV-CA	Precision Glass & Optics Contact: Thad Sanchez, thad@pgo.com Precision Glass & Optics Contact: Thad Sanchez, thad@pgo.com	75mm x 75mm x 5.5mm opal glass diffuser 75mm x 75mm x 5.5mm opal glass diffuser *(only acceptable if used as a conversion from previous programs - if used, must be only a separate production line)*

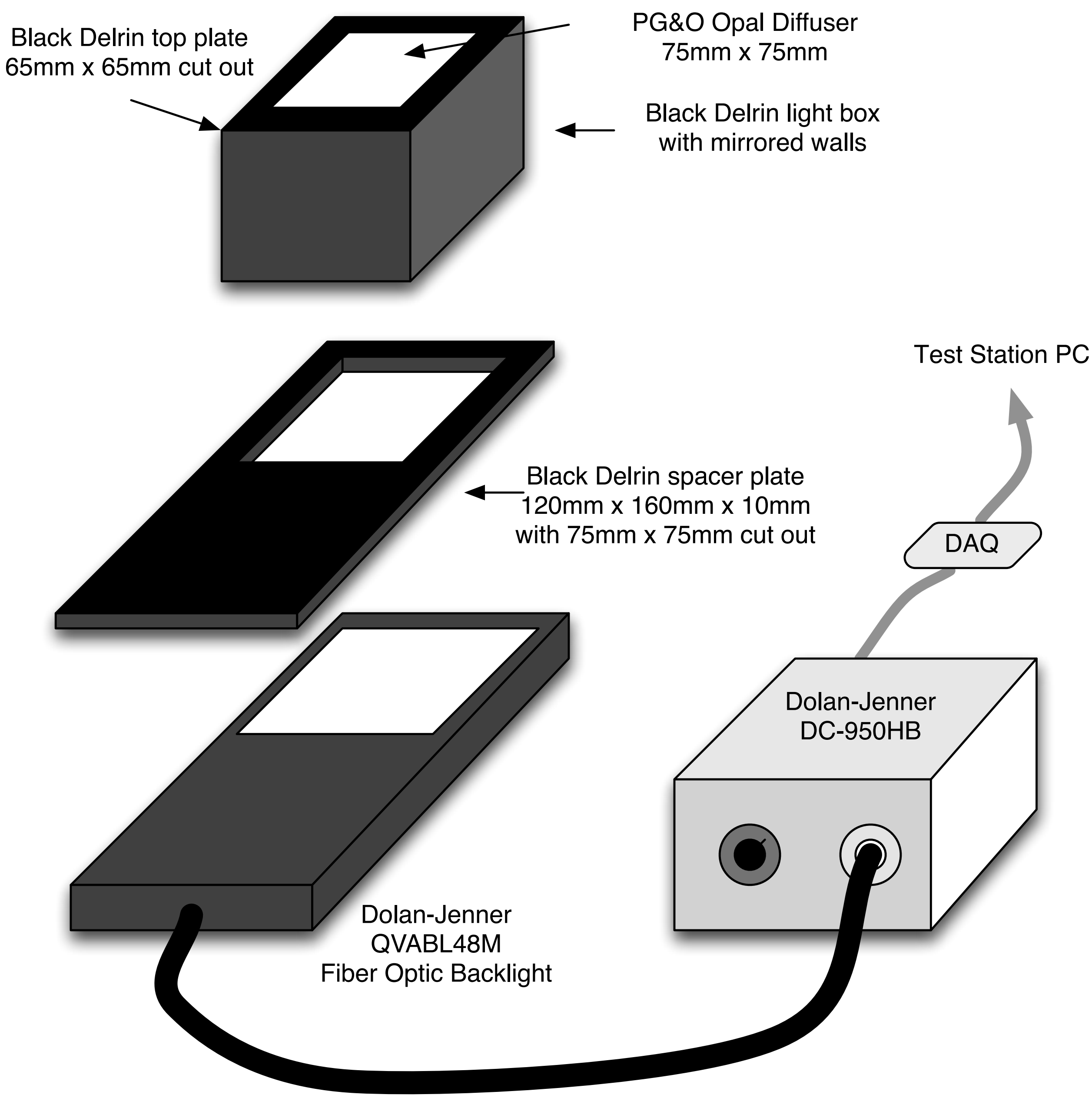


Figure 1: System Diagram of A-light (2800K)

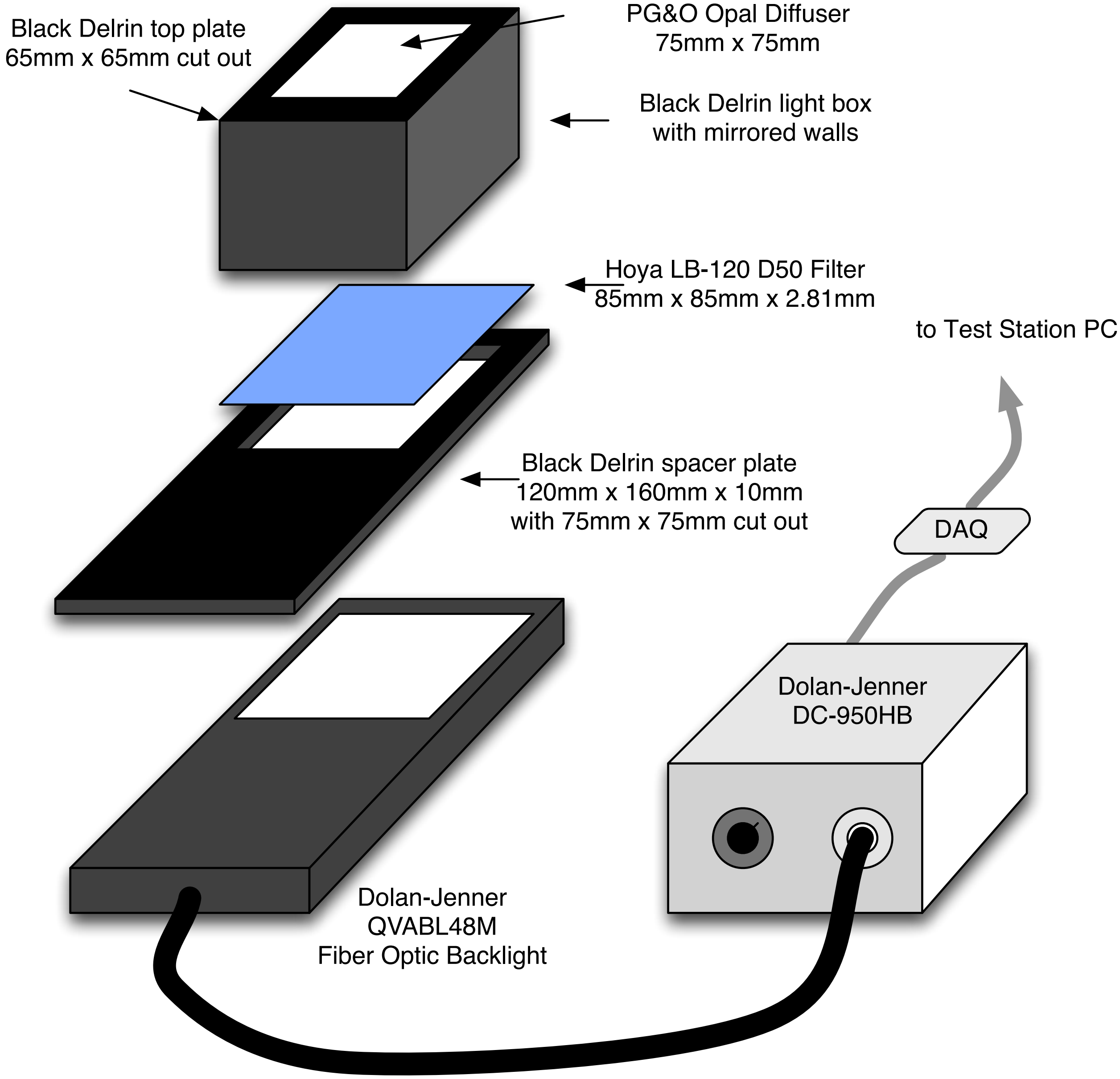


Figure 2: System Diagram of D50-light (5000K)

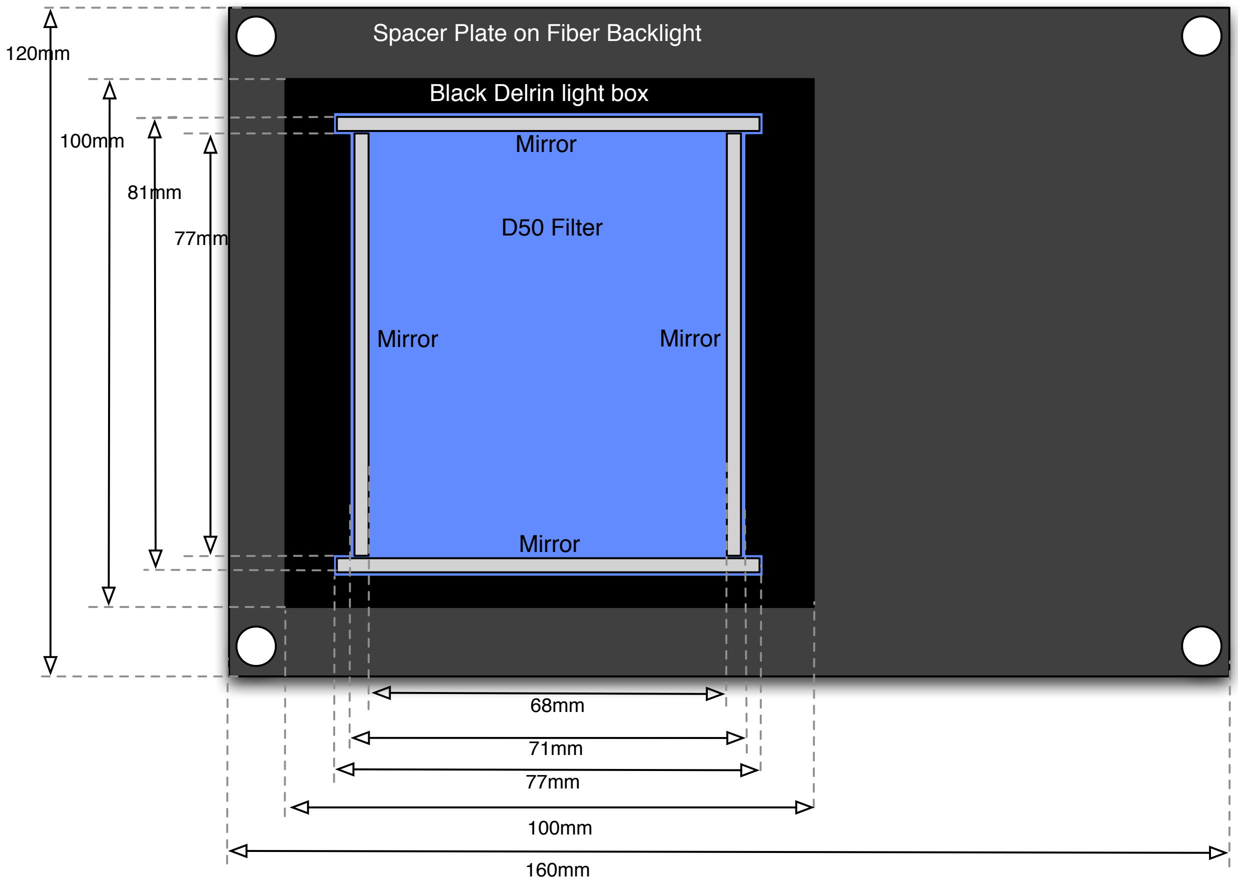


Figure 3: Fiber Light Box Top View

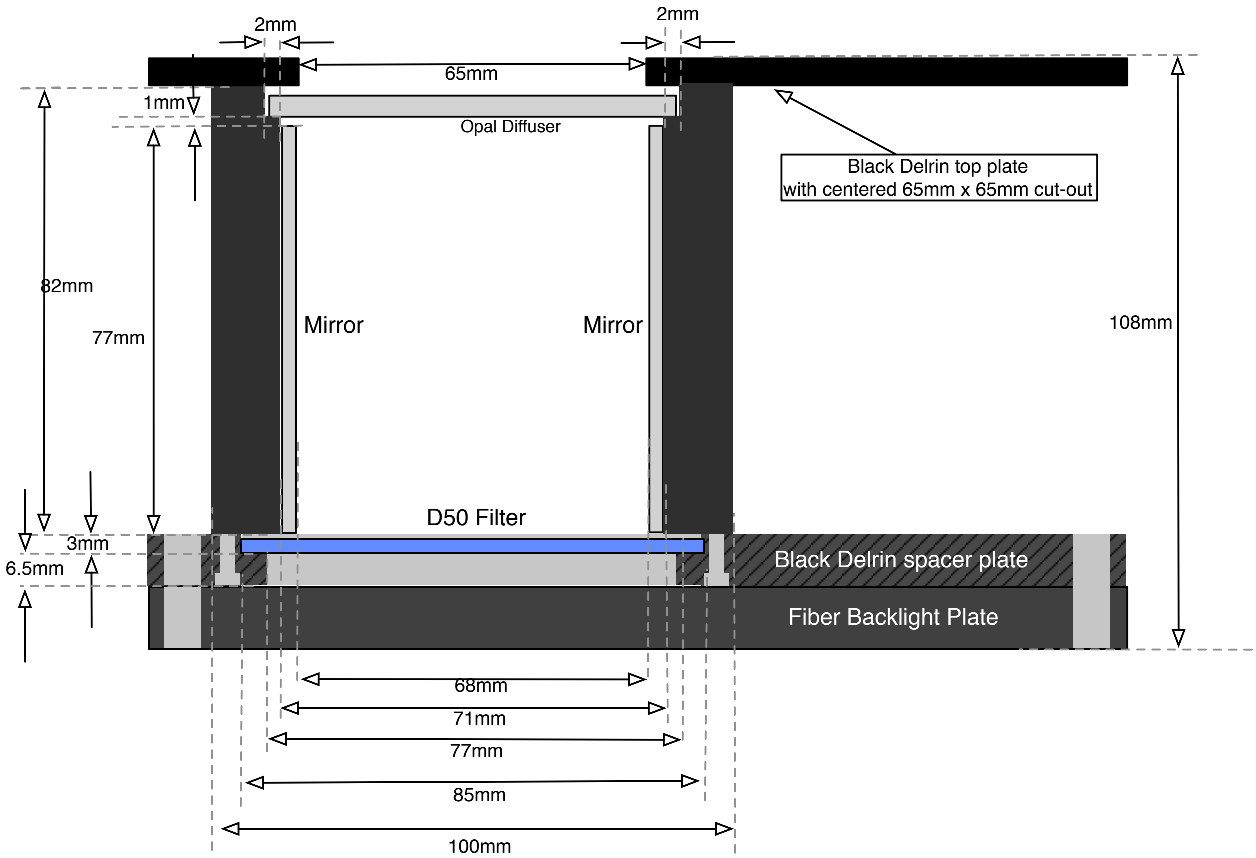


Figure 4: Fiber Light Box Side View

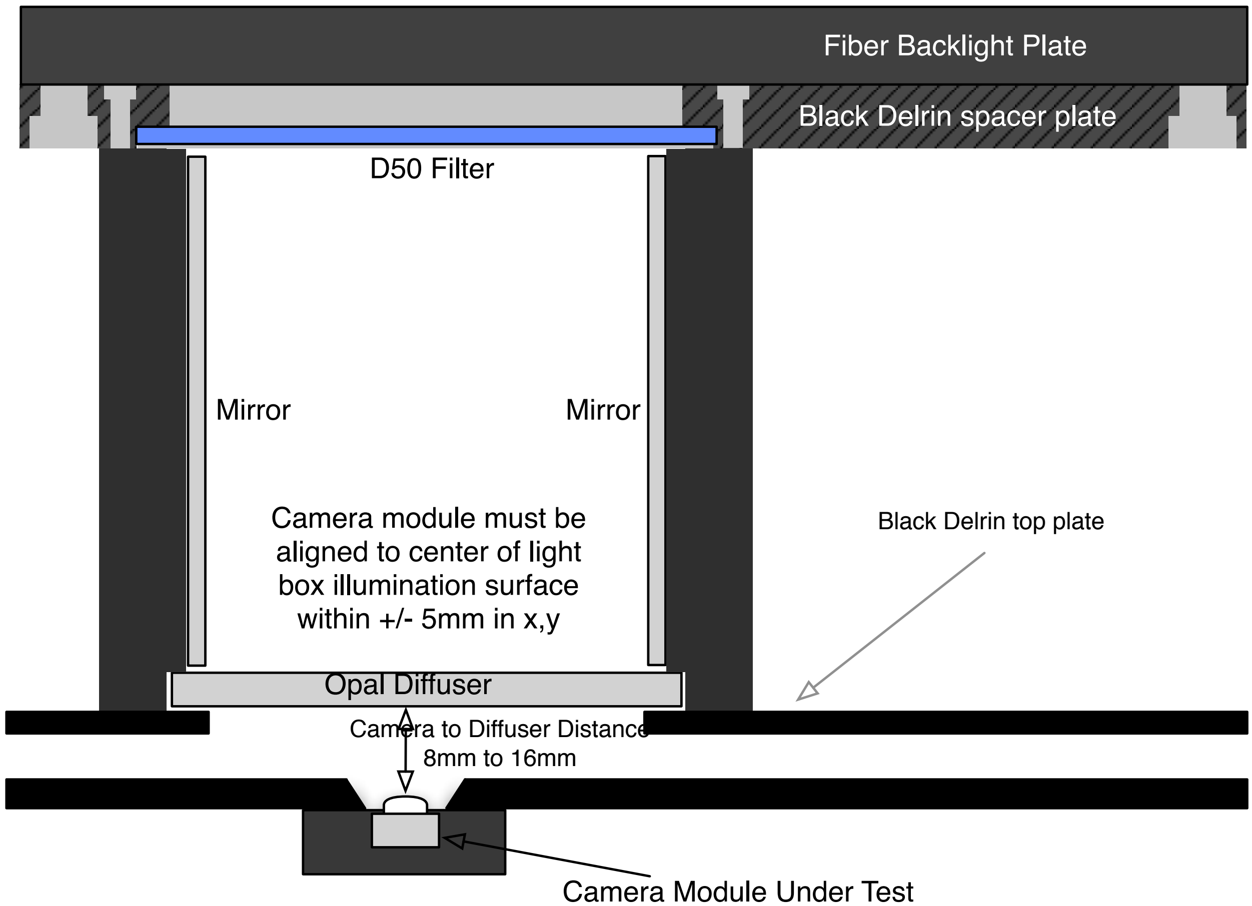


Figure 5: Fiber Light Box Side View with Camera Module Fixture

Material Inspection and Cleaning

Dolan-Jenner DC-950HB Inspection

Open front panel where filter holder and halogen bulb are located. Inspect to ensure IR filter (1" glass IR filter) is not installed. Only one glass filter should be present next to the bulb; this is a UV cut filter (2" diameter). If the IR filter is installed, remove the filter and holder. In the back of the DC-950HB, set the switch in the UP position: to Voltage (Remote).

Replace the default bulb with a long-life pre-burned EKE-XPB bulb before installing the fiber backlight.

Replace the bulb every 2 months = every 60 days.

Replace the "quick disconnect lamp socket" (bulb holder) every 2 years.

Fiber Backlight QVABL48M Inspection and Cleaning

Inspect each fiber backlight to ensure no severe scratches or cracks are present in the diffuser plate. If there are defects return to the manufacturer. The backlight diffuser surface may have surface dirt/dust when delivered so should be cleaned before installing into a new light box.

Mirror Inspection and Cleaning

Inspect each mirror to ensure no severe scratches or cracks are present. Reject mirrors with such defects. Mirrors will likely have surface dirt/dust when delivered so should be cleaned before installing into a new light box. The mirrors should be free of fingerprints after cleaning.

Opal Diffuser Inspection and Cleaning

Inspect each diffuser to ensure no severe scratches, bubbles, or cracks are present. Reject any diffusers that have defects in the centre 25mm x 25mm area since the defects will affect uniformity of the source. If diffusers have surface dirt/dust then they should be cleaned before installing into a new light box.

HOYA LB-120 Blue Filters Inspection and Cleaning

Inspect each LB-120 filter to ensure no cracks are present. Reject any diffusers that have cracks since they will affect uniformity and CCT of the source. Filters may have surface dirt/dust when delivered so should be cleaned before installing into a new light box.

Construction

Follow the light source design dimensions and specifications as outlined in the Fiber Backlight System Diagram sections of this document.

Install the mirrors with the reflective surfaces facing into the box.

Ensure the opal diffuser is installed with the smooth opal flash surface and serial number facing toward the camera module.

Ensure the fiber optic bundles have a minimum bend radius of 8cm in the system and are not under stress.

Ensure design does not allow ambient light to enter the light box or diffuser surface when camera module fixture or i1 Pro calibration meter fixture is placed on the light source.

Ensure camera module and calibration meter fixture material is matte black, spectrally neutral. Make sure the fixtures do not have any specular reflections from components such as screws or coloured material like PC boards that may effect the colour temperature and uniformity of the light source at the diffuser interface. All such components must be covered by neutral black material.

It is preferred to have the light source face down and the camera module face up in the colour calibration station to avoid dust and particles from falling on the diffuser surface of the light source. If this is not possible, then there must be a procedure to periodically clean the surface, such as using compressed air, to remove the dust and particles.

The halogen illuminator must be positioned upright, with the four rubber feet facing down.

Airflow around the illuminator should not be obstructed. Don't cover the vents and leave 3" clearance in the back.

Don't run the illuminator for extended periods with the fiber bundle removed. The lamp will cool excessively.

To change bulbs, use the software function switchOffLight() to turn the lamp intensity down to the minimum value without switching off the supply. This will keep the cooling fan running and allow a more rapid replacement.

If the ambient temperature is kept below 40C, the unit will not overheat. At higher temperatures, it may shut itself down.

Supply shutdown - The DC-950 has a crowbar circuit that is triggered when the bulb current exceeds about 9 amps. In order to reset the circuit, AC power must be removed by either removing the power cord or by extracting the bulb fixture (which opens the interlock). The crowbar circuit can not be reset by the front panel power switch.

Initial Testing and Error Handling

To initialize the light sources, run the light source calibration program GUI. Each light source calibration is good for 24 hours and must be re-calibrated again after that period.

Ensure the CCT and x,y values are within design specification. Then assign a serial number to the light source unit and record the measured CCT and x,y values in a spreadsheet file along with the given serial number. This log file of initial light source bring-up and testing should be provided to Apple for all light sources used in production. If there is an error, consult the following error handling table.

Error Code	Description	Procedure
eLSC_NO_ERROR	No error	Proceed with test
eLSC_SPECTRO_NOT_CONNECTED	i1Pro Spectrometer is not connected to the computer	Connect the i1Pro spectrometer to the computer, or try different spectrometer
eLSC_SPECTRO_NOT_CALIBRATED	i1Pro Spectrometer is not calibrated	Place the i1Pro Spectrometer on its calibration fixture (on the white calibration tile) and calibrate using *calibrateSpectro* API function
eLSC_INVALID_SPECTRO_CALIB	Error during calibration of spectrometer (i1Pro Spectrometer has to be on its calibration fixture during its calibration)	Place the i1Pro Spectrometer on its calibration fixture (on the white calibration tile) and calibrate using *calibrateSpectro* API function
eLSC_SPECTRO_SENSOR_SATURATION	Measurement out of spectrometer range (usually a too bright light source)	A light: Replace bulb D50 light: Check that D50 filter exists, then replace bulb
eLSC_LIGHT_SOURCE_TOO_BRIGHT	Light source is too bright, spectrometer might saturate	A light: Replace bulb D50 light: Check that D50 filter exists, then replace bulb
eLSC_LIGHT_SOURCE_TOO_DIM	Light source is too dim, system or bulb might be defective	Make sure fiber is attached to light source, light source is switched-on, in '*remote*' mode, light bulb is working and i1Pro is pointing at light source correctly positioned on its fixture
eLSC_LIGHT_BULB_FAIL	Light bulb failure (signal from lamp)	Check/replace light bulb Check DAQ to light source connections
eLSC_INVALID_CCT	Invalid value found while performing light source calibration (possible problem with light source, ex: A light instead of D light)	Make sure spectrometer is correctly positioned on its fixture and is pointing at the right source, examine source construction (Hoya filter only in D50 light, not in A light)
eLSC_AUTO_CALIB_MAX_ITER	Automatic calibration has not converged within the allowed number of iterations	Make sure fiber is attached to light source, light source is switched-on, in '*remote*' mode, light bulb is working and i1Pro is pointing at correct light source (either A or D50)
eLSC_DAQ_ERROR	Error while driving DAQ	Check if DAQ is correctly connected to the computer

System Calibration

Calibration of the fiber backlight system should be performed under the following circumstances:

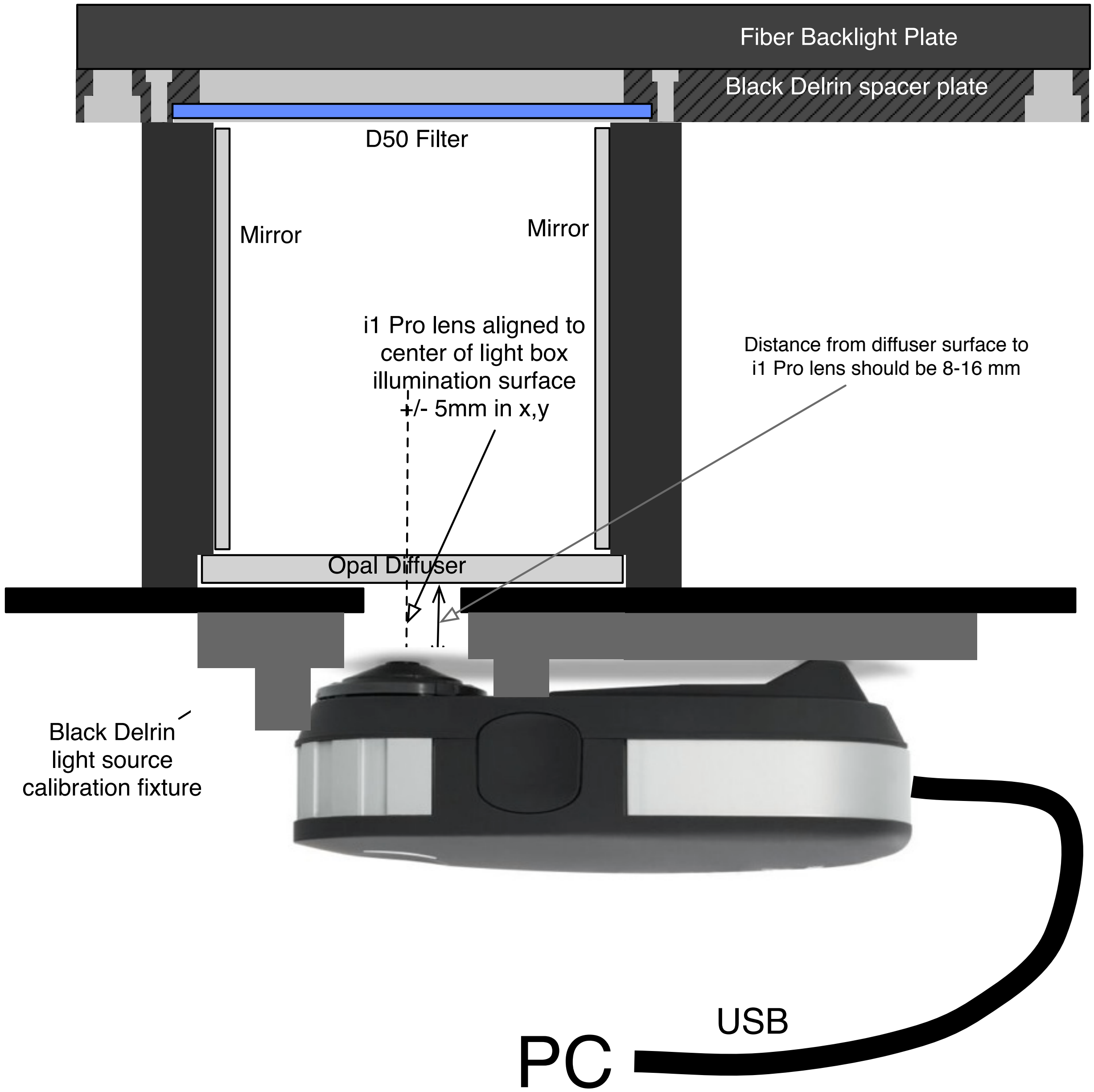
- During initial build and testing for new station
- Every time the light source is switched on
- At each bulb replacement (every 2 months = 60 days)
- At least once every 24 hours

Calibration should be performed using an X-Rite i1 Basic Pro 2 (i1Pro2) spectrophotometer along with the integrator's light source calibration program GUI. The i1Pro2 measures the CCT, x and y chromaticities, and the spectrum. The integrator software specifies the location of the diamond file provided by Apple. The Apple DLL reads the diamond file and uses it along with the spectrum to create the Residual Error Correction (REC) factors -- these must be stored on the PC and used to scale the raw colour ratios during module calibration.

The diagram below shows the placement of the i1Pro2 meter on the light box. A fixture should be fabricated that can hold the i1Pro2 to allow accurate and repeatable placement of the meter to ensure low calibration error. The fixture should hold the i1Pro2 so that the lens is within 5mm of the diffuser centre in x and y, and between 8-16mm in z. **The i1Pro2 must be connected to a PC at all times to maintain constant operating temperature. The PC must be running the light source calibration GUI to maintain a USB connection to the i1Pro2.**

Fiber Backlight System Calibration Equipment List

	Part Number	Manufacturer	Description
1	i1 Basic Pro 2 EO2BAS	X-Rite USA Phone: 1-800-248-9748 http://www.xrite.com	Handheld USB spectrophotometer for measuring and calibrating CCT, x and y, and spectrum of light source. Includes driver software. Handle with care, especially the diffuser head sensor. Keep the unit in the white-dot calibration plate when not in use. Return the device to X-rite every year for recalibration.
2	NI USB-6008	National Instruments Check with local NI offices or distributors	USB Analog/Digital data acquisition (DAQ) See software documentation for wiring diagram
3	Light Source Calibration GUI	Integrator	Software to control DAQ and set/hold light source voltage, use with Apple DLL and i1Basic Pro 2 software driver



The i1Pro2 must be connected to a PC at all times to maintain constant operating temperature.

The PC must be running the light source calibration GUI to maintain a USB connection to the i1Pro2.

Figure 6: System Calibration Placement Diagram

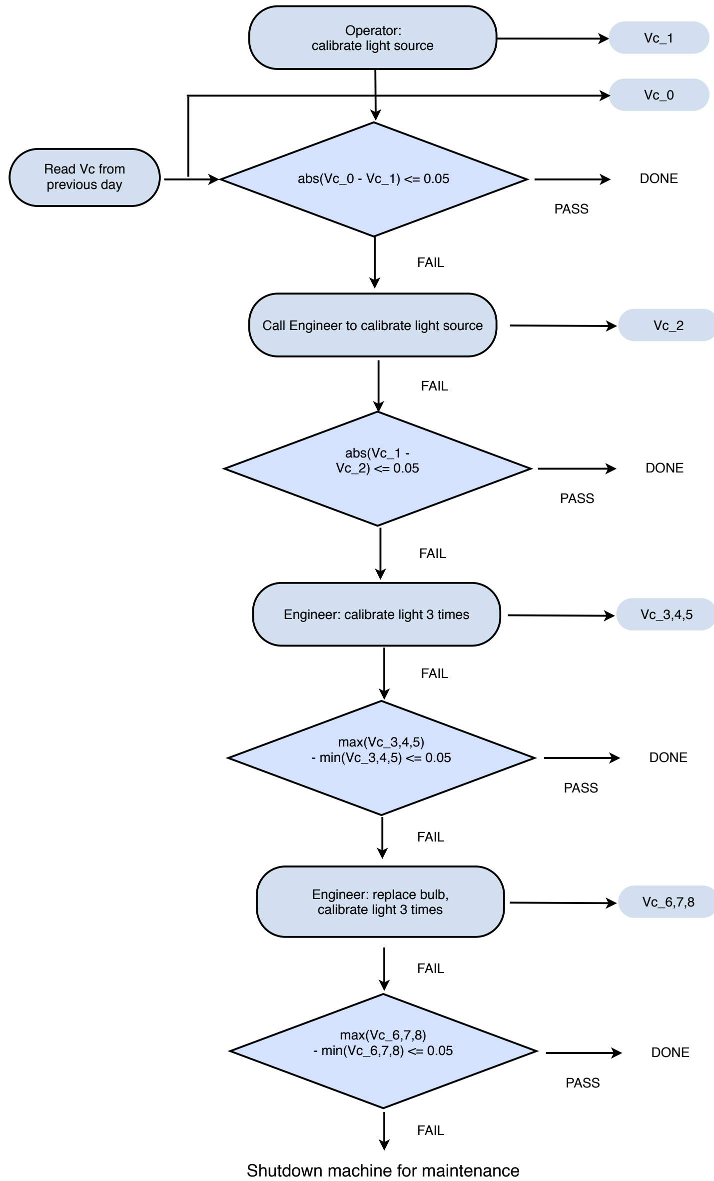


Figure 7: System Calibration Methodology Flowchart

Appendix B: Change History

Version	Description	Date	By
1	Initial rollout	1/20/2013	Eugene Lam
2	- increased image quality - general spelling and grammar modifications - light requirements now specified as luminance and colour temperature ranges - updated preprocessing indexing methodology - dark light requirements updated to <1lux - colourShading configuration table no longer mentions use of manual WB - removed reference to per module lens shading from colourUniformity test flowchart - electrical standby current condition specified as hard standby - fpn data is reported per colour channel - read noise calculates mean of deltas as opposed to std dev - relativeUniformity field neighbour delta no longer normalizes the values - SFR data centre changed case to lower case - SFR delta data renamed to signify subtraction is right - left side - temporalNoise formula adjusted to report std dev of means - LCB/LCBraw adjusted so an absolute threshold is used to determine severe defects that should not be removed through filtering (previously this was done relative to the limit) - all captures standardized to 5 averaged, except for SFR and temporalNoise - actuator test naming modified from VCM test - NVM map reference now points to module ERS - Appendix A added for light source specification - Appendix B added for change history	3/22/13	Eugene Lam
3	- preprocessing AWB description corrected for corrWB calculation - defective pixel test added in ARPD and features - defective line test output for dark revised to be LSB units - defective line test dark condition revised to be minimum integration time, maximum gain - fixed pattern line test removed reference to light field - SFR now includes AWB and LSC	4/4/13	Eugene Lam

Version	Description	Date	By
4	minor formatting changes - Authority section added to general comments - all references to colourCal documentation now refer to Appendix A - all references to 16x gain replaced with maximum sensor allowable gain - all dark noise metrics aside from readNoise now specify a 66ms integration time - colourCalibration now includes Test Image section - colourCalibration light source calibration requires 5min startup time only if a power off occurred - colourUniformity test applies per module LSC (as opposed to a generic one) - contrastTest specifies the use of manual LSC - defectivePixel test added border detection - defectivePixel test ARPD check now explicitly states it is only required for light field - no change to actual algorithm - defectivePixel test typo corrected detectability (sign change) - no change to actual algorithm - LCB threshold replacement modified to specify the max of either the horizontal or vertical filter (documentation update, algorithm has remained the same) - SFR auto-exposure explicitly targets the y-channel - actuator testing updated to reflect newer test methods - flat field light source can use a 3mm diffuser if it was converted from a previous program - flat field light source A light expected range increased to 100-400 cd/m2	5/29/13	Christos Hristovski Eugene Lam
5	- tester and socket information to be stored in the log file - D50 flat field images now have a specified lens position - defectivePixel test NVM defect removal excludes border pixels - defectivePixel test border pixels now from DPs or DPPs - updated DSNU algorithm to be more sensitive, reduced warm-up time to 2.5s - new TemporalNoise algorithm - updated SFR documentation to show all ROI - updated Defective Pixel documentation and test to new priorities - updated diffuser part number to 0075-0075-0060-GV-CA due to PGO labeling issue, material is the same	11/09/13	Christos Hristovski Eugene Lam

Version	Description	Date	By
6	- added requirement to log sensorID - A light testing to be done with lens at infinity position - relativeUniformity testing is done at max bit depth (previously stated 10-bit)	5/15/14	Eugene Lam