



Raw to Real,
Green to Great

FOSDEM 2026

goeiedag, bonjour

- Partner at [Ideas on Board](#)
- Embedded Linux software developer
- 20+ years of experience with [Embedded Linux](#)
- ~10 years of experience with Cameras
- Release manager, Co-Maintainer, Support for [libcamera](#) project
- Linux kernel contributor, component maintainer



kieran.bingham@ideasonboard.com
902D D798 3FBD 28FD E372 D47A A11E 4672 10A4 61FD

About me



Kieran Bingham
Director of Engineering
Bristol, UK
kieran.bingham@ideasonboard.com
www.ideasonboard.com

Open sourcing the tuning process

- Problem statement
- The process
- A proposal
- Help!

There's a lot which *can't* be covered in this small slot for future discussions (De-noise, 3D-LUT, Edge enhancement, sharpness ...)

Agenda



GraphView Histogram

Parameter	Value
Controls	
Contrast	1.0
Saturation	1.0
Metadata	
AnalogueGain	9.578125
ColourCorrectionMatrix	(2.0, 0.0, 0.0, 0.0, 1.0,
ColourGains	(1.386798620223999
ColourTemperature	5672
DroppedFrames	0
ExposureTime	33600
FrameRate	29.952674773857307
FrameSequence	299
Saturation	1.0
SensorBlackLevels	(4096, 4096, 4096, 4
SensorTimestamp	8657737130000
Info	
Color	R: 234 (91.8 %), G: 18
Format	SGRBG10P
Mean	196 (76.9 %)
Size	1296x972
Overlays	
› Crop Detector	
› Mac Beth analyzer	✓
Properties	
ColorFilterArrangement	1
Location	0
Model	ov5675

+

-

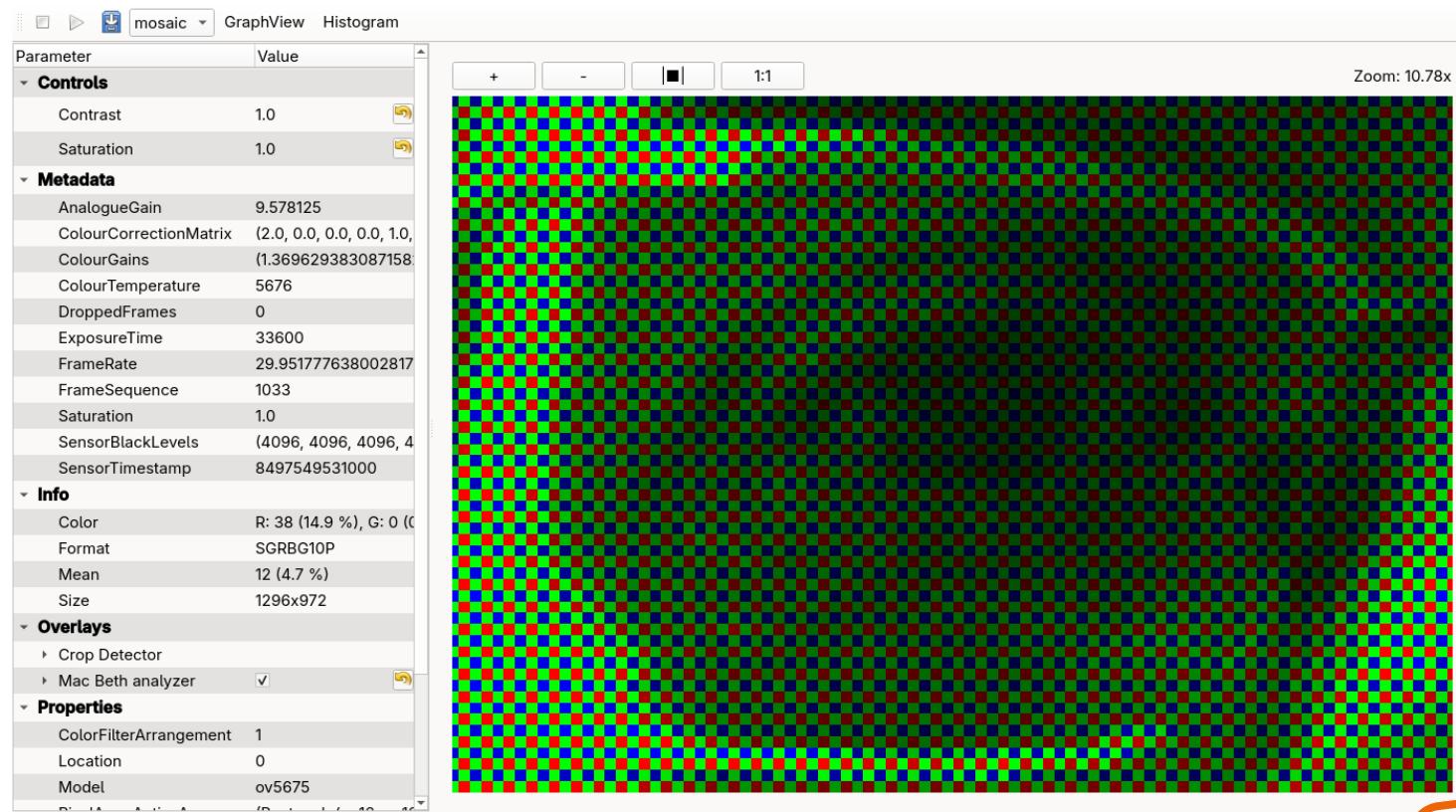
1:1

Zoom: 0.74x

R: 234 (91.8 %), G: 182 (71.4 %), B: 173 (67.8 %) Lum:130 P:4:

RAW Green





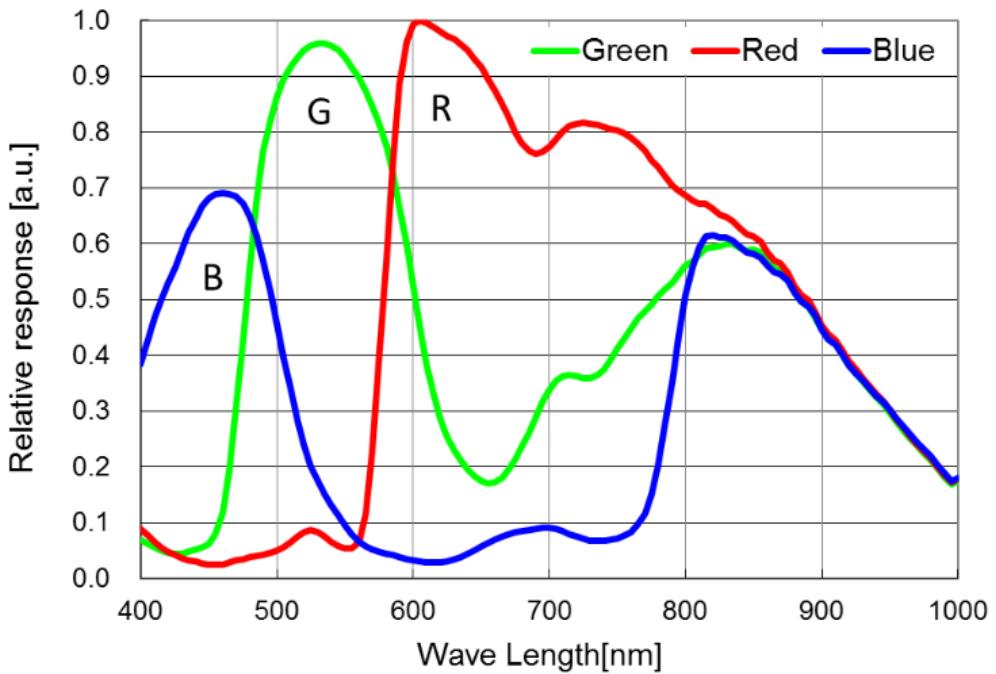
RAW Bayer / Mosaic



Spectral Sensitivity Characteristics

(Excludes lens characteristics and light source characteristics.)

*Graph taken from a sensor datasheet



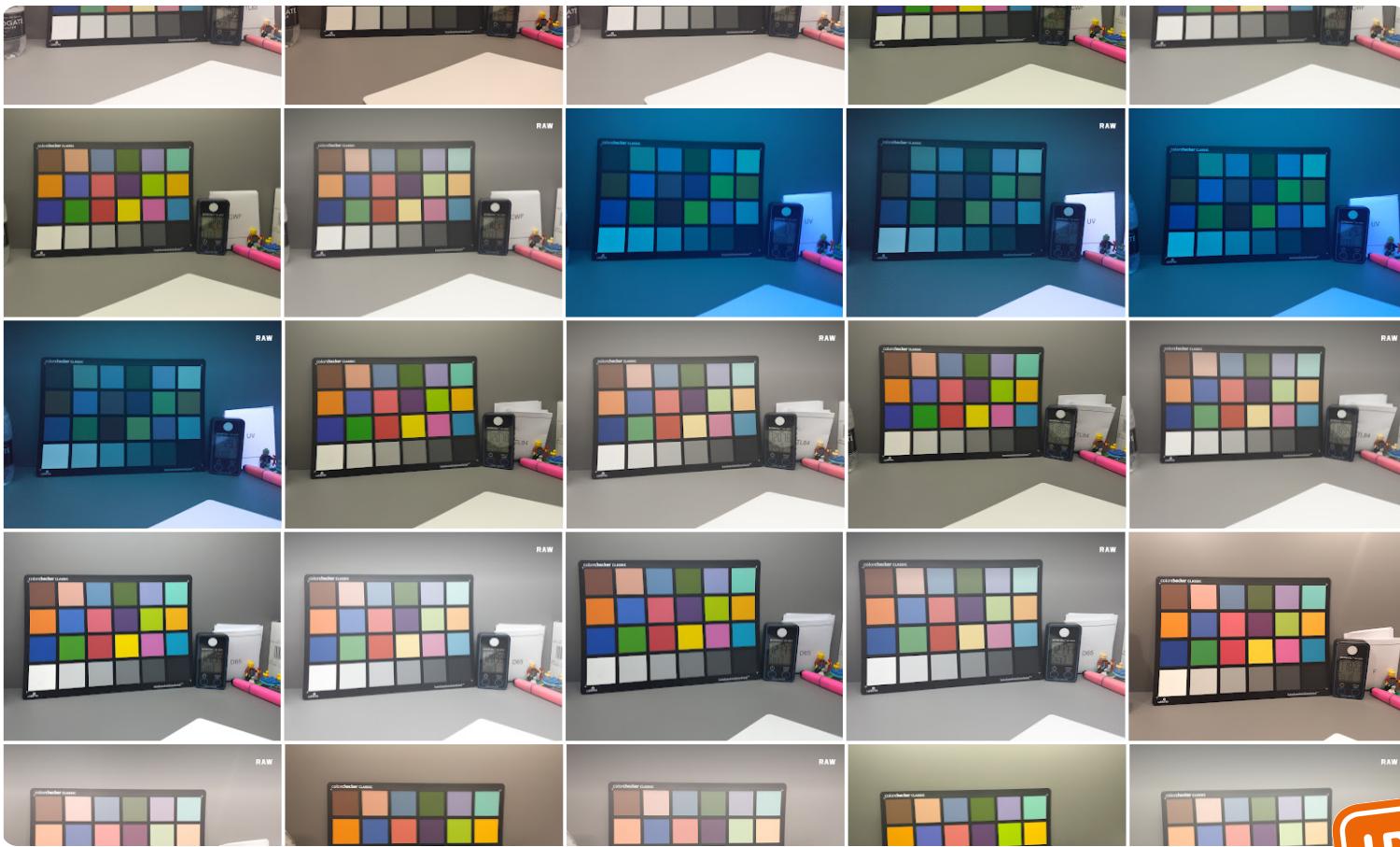
Spectral Sensitivity





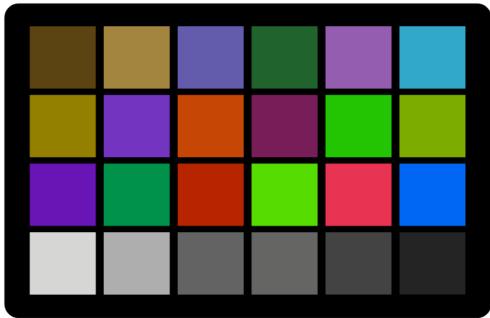
Calibration Light Box

IDEAS
ON BOARD

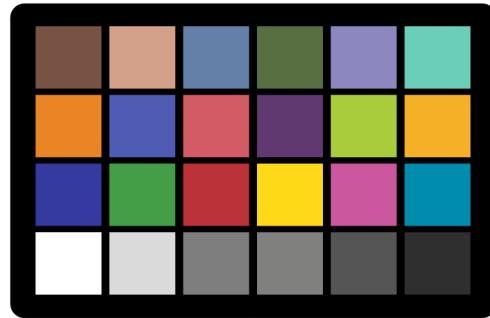


Capture

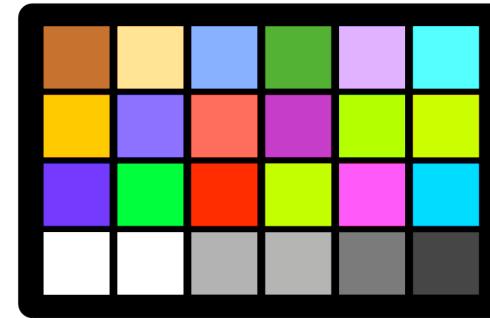
IDEAS
ON BOARD



3200k



4000k



6400k

Colour Charts to Delta-E



	R'	G'	B'
R	2.121	-0.525	-0.596
G	-0.853	2.804	-0.951
B	-0.269	-1.148	2.417
2860k			

	R'	G'	B'
R	2.182	-0.709	-0.473
G	-0.702	2.764	-1.062
B	-0.252	-0.720	1.971
4650k			

	R'	G'	B'
R	2.212	-0.532	-0.679
G	-0.579	3.079	-1.500
B	-0.277	-0.733	2.010
7580k			

- Real CCM values from the IMX219

Colour Correction Matrix



	R'	G'	B'
R	2.022	-0.419	-0.603
G	-0.595	2.215	-0.620
B	-0.440	-1.169	2.609
2676k			

	R'	G'	B'
R	1.939	-0.675	-0.264
G	-0.430	2.007	-0.577
B	-0.157	-0.513	1.670
4650k			

Fairphone 5 first pass camera CCM





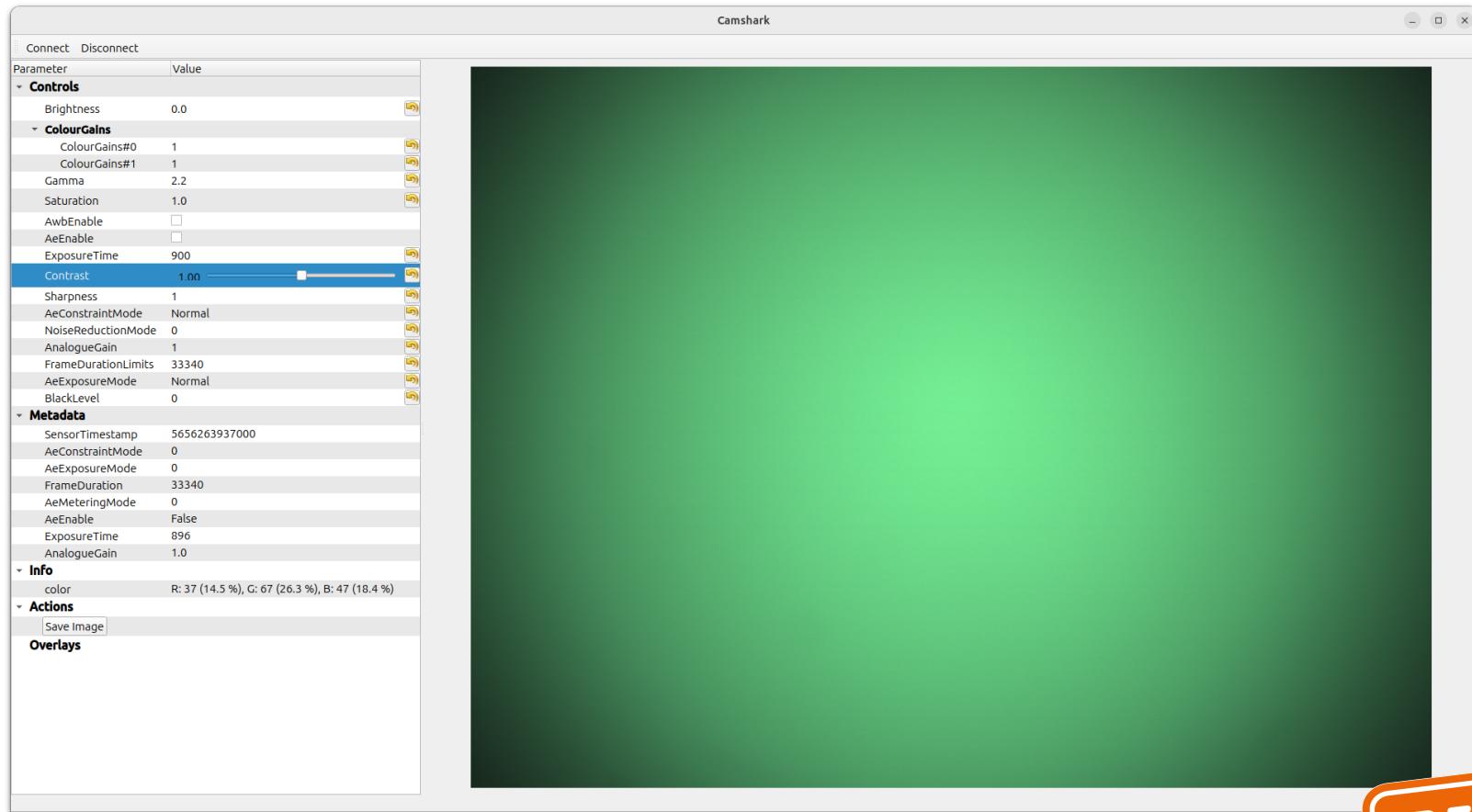
- Lens shading under development
- AWB should move to AwbBayes
- CCM / Cross-talk

Future:

- Autofocus
- Denoise/Sharpen
- 3D LUT
- Lens distortion correction

Simplified (GPU) ISP Pipeline

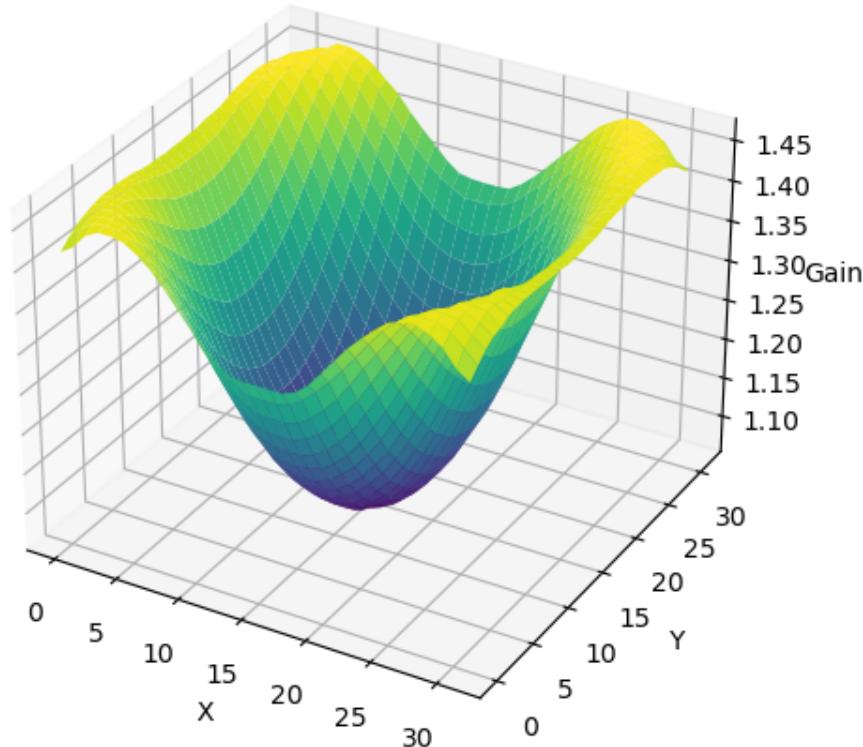




Lens Shading



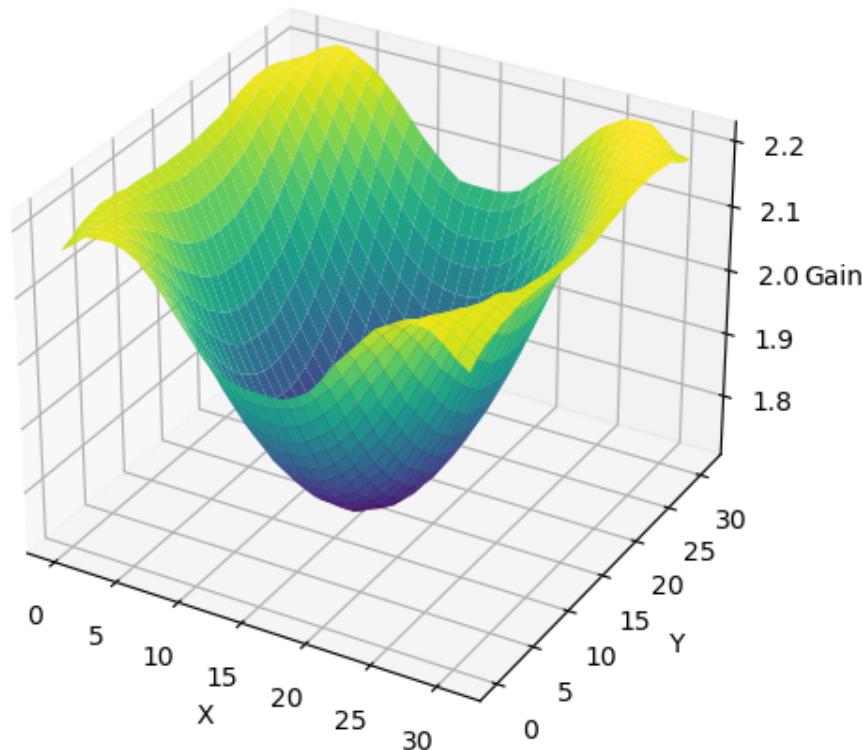
ALSC Shading Table at CT=3000K



Lens Shading



ALSC Shading Table at CT=5000K



Lens Shading



```
# install the necessary python packages
cd libcamera
python -m venv venv
source ./venv/bin/activate
pip install -r utils/tuning/requirements.txt

# run the tuning script
utils/tuning/rkisp1.py -c config.yaml -i ./tuning-data/ -o tuning-file.yaml
```

- Parses directory
- Examines filenames
- Should use metadata

Tuning Tool





Camera Algorithm and Tuning Guide

The Raspberry Pi Camera Algorithm and Tuning Guide

Essential Reading

67 pages of gospel from David and the team

- <https://datasheets.raspberrypi.com/camera/raspberry-pi-camera-guide.pdf>

Stefan is working on guides for other platforms

- <https://libcamera.stefanklug.com/guides/tuning/tuning.html>
(will be merged to libcamera.org)

Gospel



- Bayer images – We start with RAW images
- But why is it green really
 - AWB + CCM (Plus others) need calibration
- Tuning a mobile phone camera can be hard
- Somewhat expensive / bulky equipment required

Problem statements



Tuning 100's of
mobile phone
cameras, and
laptops, and
tablets and ...

THE BIG PROBLEM



70%

Minimum Equipment needed

- Colorimeter - **£50**
- Colour checker card - **£80**

80%

Custom light box / Self builds

- Ikea box + light automation - **~£200**
- Flat panel LED - **£30**

90%

Extended equipment:

- Full light box - **£800**

100%

Commercial lab: **£100,000 +++**

Equipment required

SK-8202 (with illuminance + color temperature + color rendering index Ra)

Illuminance measurement range:
0.5-300000LUX/ 0.02-150000FC
Maximum display: 999999
Measuring gear: 999999/300000
Resolution: <10000.0, 1.0, >10000.1
Illumination accuracy: \pm (4%+10)
Illuminance repeat test: \pm 2%
Color temperature CCT: 2500K-9000K (\pm 5%)
Color rendering index Ra: 20-100Ra (\pm 2%)
Sampling rate: 4 times / sec
Incidence angle: 100 degrees
Spectral range: 400nm-700nm



Digital Illuminance Meter Light Meter Handheld Lux Tester 0.5-300,000 Lux/Fc Pocket Luxmeter Photometer Grow Plants Luxometro

★★★★★ 4.2 12 Reviews | 41 sold

£49.79 50% off £99.58

Price includes VAT

Pay with PayPal Pay in 3 interest-free installments

Color: SK-8202



Calibrite ColorChecker Classic: Colour Chart for Photography and Filmmaking
Visit the Calibrite Store

4.5 ★★★★☆ (96)

-11% £79.00

RRP: £89.00

Style Name: Classic

- The original ColorChecker for colour calibration and control from capture to edit
- Colour chart for true-to-life colour reproduction for photographers and filmmakers when checking images and cameras, correcting white balance and performing colour correction
- The standard 24-patch target, A4-sized, for all users where colour control is key
- 6 greyscale patches, RGB & CMYK colours and a mix of everyday colours for great results
- Use this important photography accessory with the ColorChecker Camera Calibration software to create ACR or ICC profiles

Report an issue with this product



Captures vary in size but the collection will grow rapidly:

- Tunings are about 500MB to 1GB or maybe up to 2 GB per camera module.

```
libcamera@tuning:~/ $ du -sh *
448M vendors/fairphone/fp5/front
421M vendors/raspberrypi/imx219
606M vendors/raspberrypi/imx708
116M sensors/omnivision/ov8858
493M sensors/sony/imx283
915M sensors/sony/imx335
418M sensors/sony/imx363
285M sensors/sony/imx415
97M sensors/sony/imx462
```

Captures



Our new problem statement

- We have lots of phones.
- Lots of laptops.
- Lots of tablets.
- Varying ISPs.
- Few people with internal knowledge of the ISP pipeline.

\$DEVICE_VENDOR has '100' people per phone.
We have '100' devices per person to support.

Our goal.

- Democratise, and expand tuning for everyone.
- Requirements. We have to make it easy.
- Make it open, so people build on top of existing tuning, not start from scratch.

Open source side



It turns out tuning is 'tricky'.

- Light box definitely provides mains flicker @ 50hz.
- 'Calibrated' lights and a cheap Snarkol SK-8201 will disagree
- It's simply 'fiddly' to get everything in the right place
- It's important to make sure you get a good exposure and gain
 - And make sure the colour checker card is clear and visible

Issues



- Try to include the light meter in the capture to be able to reference in pp
- Include the light 'name' in the scene
 - Printed paper worked for me
 - F, D65, TL84, UV, CWF, TL83
- Use a 'dark' room or capture at night to avoid 'light pollution'
- Make scripts to help save out the capture files
 - Or later we should make assisted capture tools
- Make sure the lights are not in shot
- Don't face the light meter *at* the lights
- Don't be afraid to delete captures!
 - You can always capture more

Tips



T tuning @

main tuning

Select Git revision

Search by Git revision

Branches 7

- raspberrypi/imx708
- raspberrypi/imx219
- sony/imx219
- fairphone/fp5
- sony/imx462
- sony/imx363

captures ...

354dc5a1

Find file

Code

Last commit Last update

Last commit	Last update
libcamera-tuning: Provide initial README...	1 month ago
.gitattributes: Use LFS for raw captures	1 month ago
libcamera-tuning: Provide initial README...	1 month ago
libcamera-tuning: Provide initial README...	1 month ago

README.md

libcamera Tuning Images Repository

This repository contains raw image captures used to develop and validate libcamera tuning files. It is intended as a shared, open resource to support camera module tuning across different sensors, lenses, and platforms.

Aims

Camera tuning requires carefully captured raw images and metadata about the scene. This repository provides:

- A structured home for raw tuning captures
- A clear workflow for generating and reviewing libcamera tuning files
- An open licensing model to enable reuse and collaboration

Large File Support

It is anticipated that this data storage could grow large, and therefore only generated tuning files and tooling should be merged on to the main branch.

All image captures should be maintained in a module specific branch with large files stored using git-lfs so that large objects are only

Start simple...

Can git help ?

- <https://gitlab.freedesktop.org/camera/tuning>

Pros and cons

- Easy ...
- 'familiar'
- Requires git-lfs
- We can't fork a 500 GB repo easily

Alternatives?

- Move to S3 buckets and a web interface

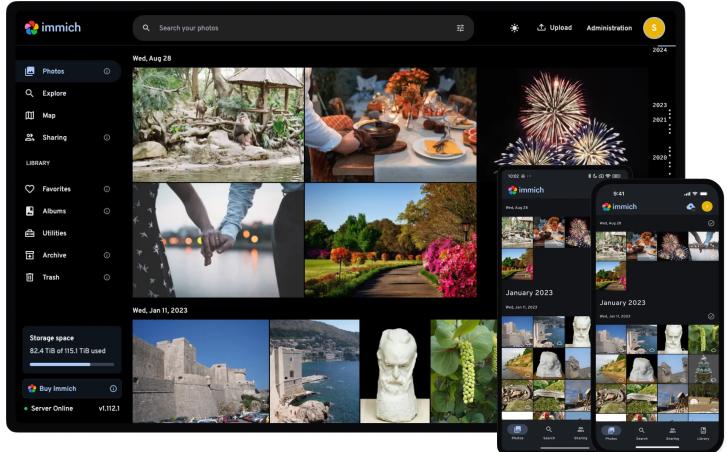
camera-tuning



Help wanted

Where do we need help...

- A web front end
 - Viewing images
 - Backend to run the tuning scripts
 - Interface to change parameters and display results
- Building on top of something existing (immich?)
 - Or do we build our own backend service entirely ?
 - What about a command line interaction directly to SoftISP



Help!



I'd also like to propose a web interface or such on top of the Soft/GPU ISP.

- Backend using the exact same shaders
- Front end supplies 'one of the raw's
- Sliders to control parameters
- Upload the image,
- Process with the parameters
- Show the result
- Calculate delta-e directly when a colour chart is detected

Your design here ...

Web ISP



Take (RAW) pictures!

- With a colour chart
- With an estimation of colour temperature
- With an estimation of lux level
- Perhaps with ground truth / android capture too?

Build infrastructure

- Storage and sharing system
- Parameter tweaking interface



<https://matrix.to/#/#libcamera:matrix.org>

Contribute





Sliddev

Presentation Slides for Developers

Tools / Thanks



Your vision. Made real. Together.

IDEAS
ON BOARD



info@ideasonboard.com

+-/ \ -+
| (o) | libcamera
+-----+



#libcamera on oftc.net

<https://matrix.to/#/#libcamera:matrix.org>



libcamera-devel@lists.libcamera.org

Thank You!

