CA378-AOIS for JetsonTX2 HDR Processing Guide

Version 1.1.3

Dated: 2018/06/15

Home Page ***https://www.centuryarks.com/***

History

Date	Version	Comment
2018/06/15	v1.1.3	Initial Release

Environment configuration

CenturyArks Co., Ltd.

Before running CA 378-AOIS software, please install the following environment.

Build and install OpenCV 3.4.1

```
$ cd ~/CA378_2L_v1.1.3_L4T28.2-rc_src_build
$ ./InstallOpenCV.sh
```

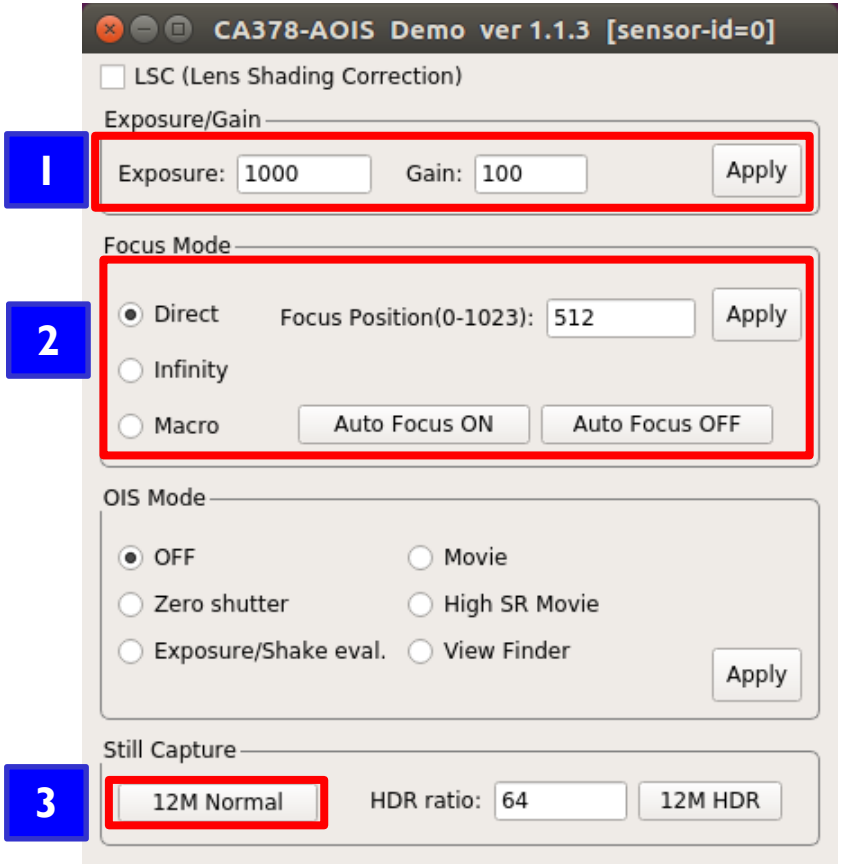
Install RAW development software

```
$ sudo apt-get update
$ sudo apt-get install ufraw
$ sudo apt-get install v4l-utils
$ sudo apt-get install geeqie
```

12M pixel still image capturing

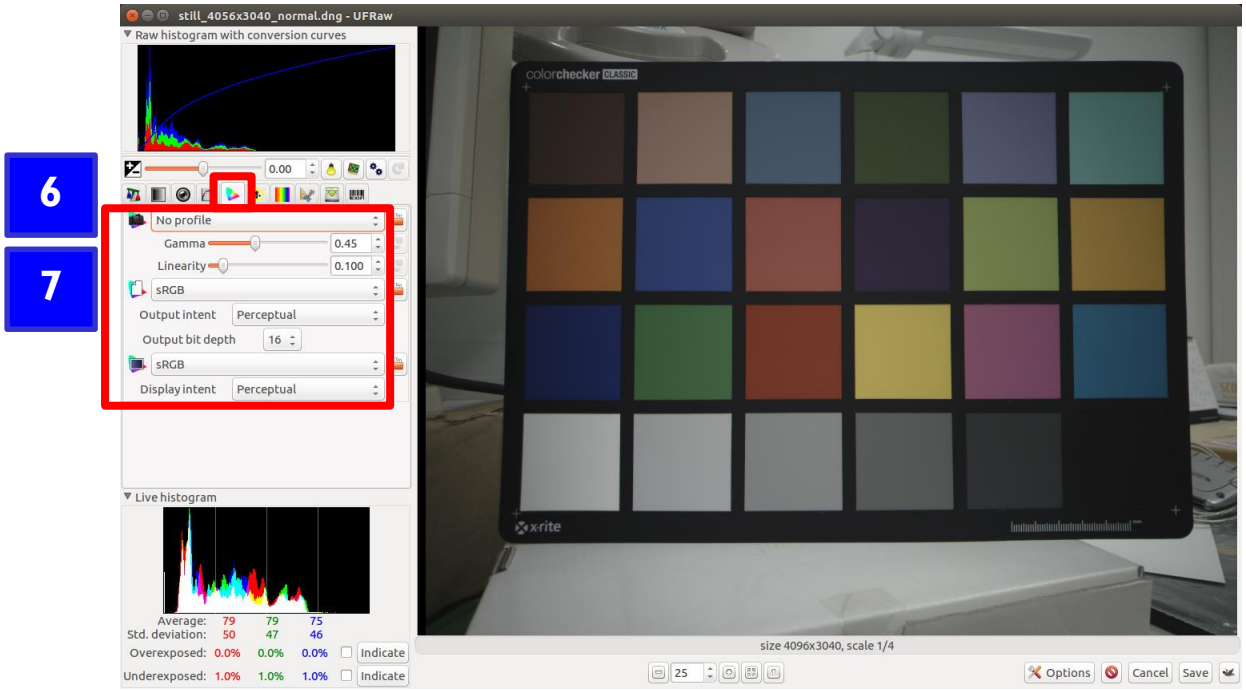
CenturyArks Co., Ltd.

Procedure of normal capturing 12M pixel still image:
1. Adjust the Exposure / Gain.
2. Adjust the focus.
   (It is useful to turn on Auto Focus and turn Auto Focus OFF when focus is on)
3. Click the [12M Normal] button



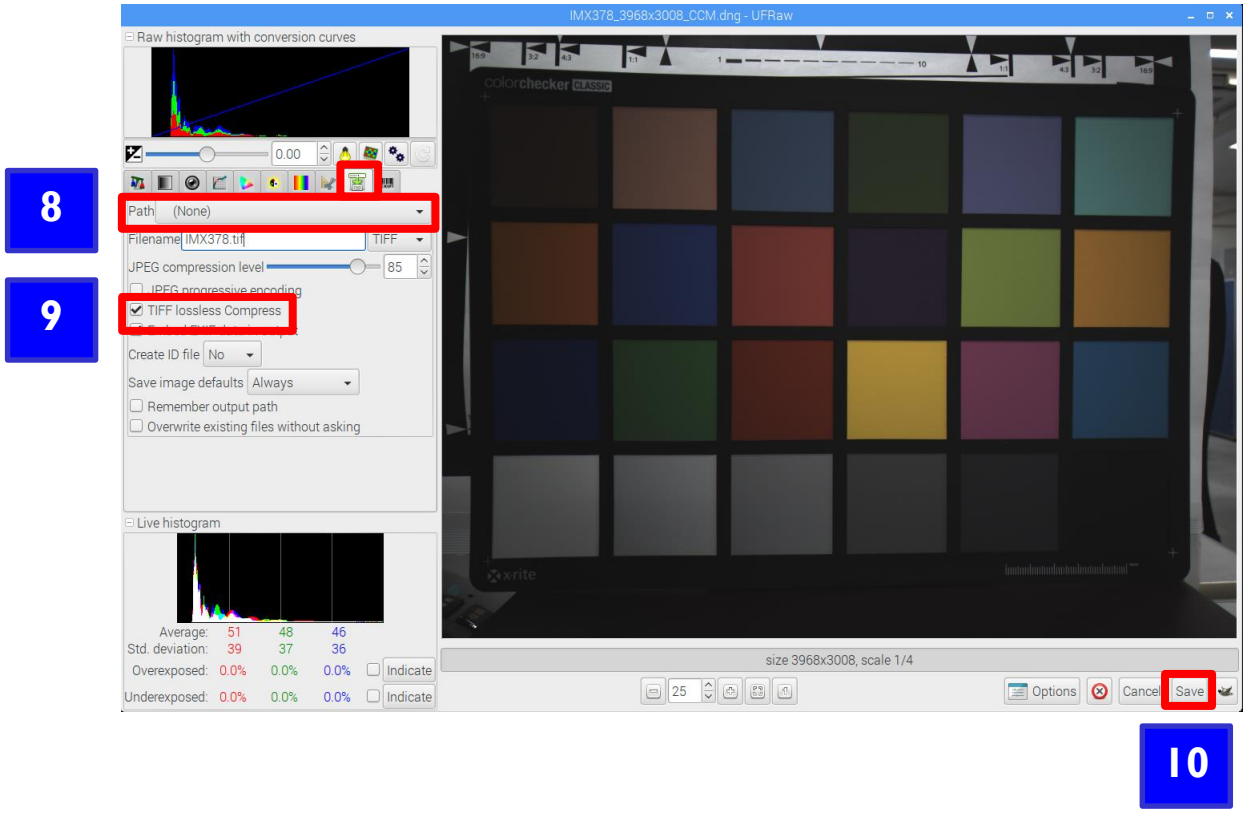
Raw configuration

5. UFRaw will be started when capturing is completed.
6. Set the camera profile of color management to "No profile".
7. Set Gamma to “0.45".
   Set Output bit depth to “16”.
   Set Display to “sRGB”.



Raw configuration

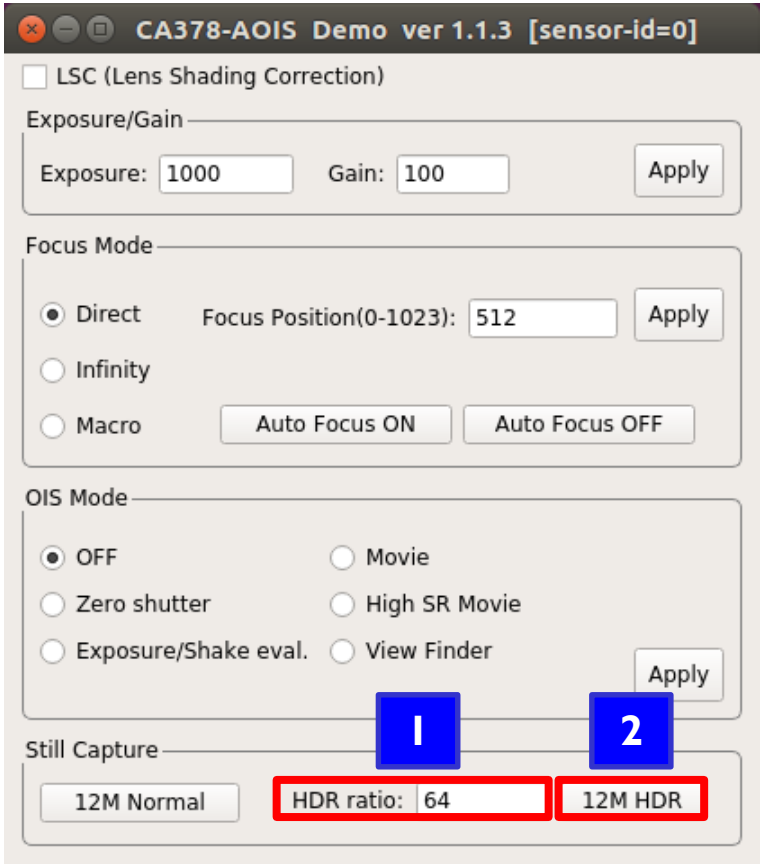
8. Change "Filename" of Save to “xxx.tif".
9. Check "TIFF lossless compress".
10. Click the [Save] button.
11. Exit UFRaw
12. Exit Demo



12M pixel still image HDR capturing

CenturyArks Co., Ltd.

Procedure of HDR capturing 12M pixel still image:
1. Adjust the HDR ratio.
2. Click the [12M HDR] button



Display in HDR image

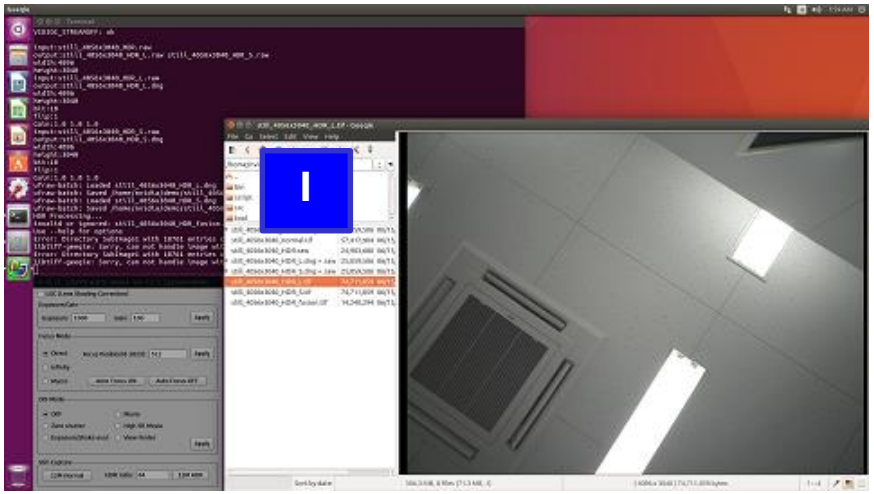
CenturyArks Co., Ltd.

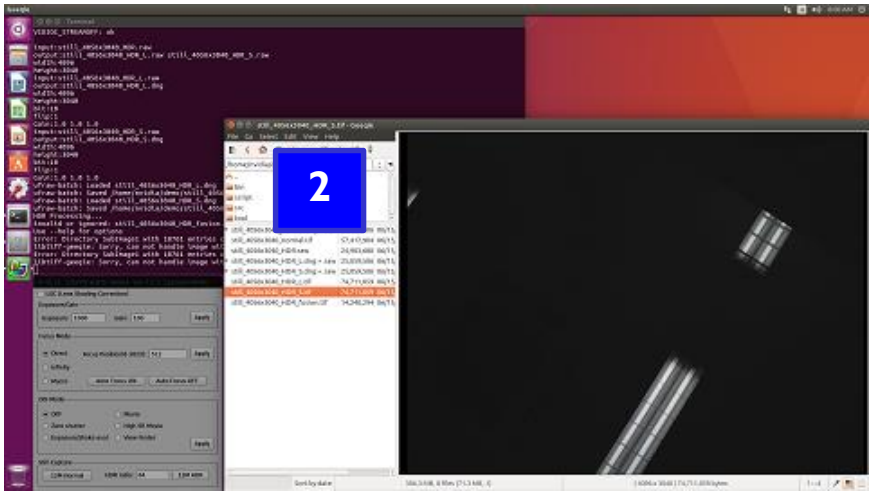
The following image is saved in the “~/demo/” directory.

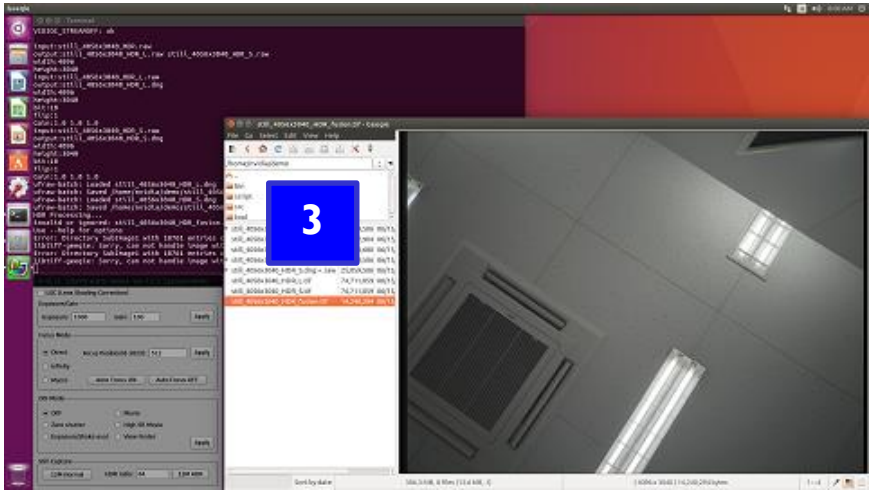
(1) HDR long exposure image

(2) HDR short exposure image

(3) HDR tone mapping image







Modify HDR tone mapping algorithm

CenturyArks Co., Ltd.

In case of modifying HDR's tone mapping algorithm, please change "~/demo/script/HDR.py":

```
# -*- coding: utf-8 -*-
import cv2
import numpy as np

print('HDR Processing...')

# Loading exposure images into a list
img_fn = ["still_4056x3040_HDR_L.ppm", "still_4056x3040_HDR_S.ppm"]
img_list = [cv2.imread(fn) for fn in img_fn]

# Exposure fusion using Mertens
merge_mertens = cv2.createMergeMertens()
res_mertens = merge_mertens.process(img_list)

# Convert datatype to 8-bit and save
res_mertens_8bit = np.clip(res_mertens*255, 0, 255).astype('uint8')
cv2.imwrite("still_4056x3040_HDR_fusion.ppm", res_mertens_8bit)
```

For HDR tone mapping algorithm see below.
https://docs.opencv.org/3.4.1/d3/db7/tutorial_hdr_imaging.html#gsc.tab=0