ELDIM
ELECTRONICS FOR DISPLAYS AND IMAGING DEVICES

Application Note: Capture Sequence

Abstract	This documents is an application note
Version	0.2
Status	Draft
Date	2020/07/23

Revision history

Version	Date	Content
0.1	2020/07/22	Initial version
0.2	2020/07/23	Add chapter about X, Y, Z images

ELDIM
1185 Rue d'Epron (Ancienne)
14200 Hérouville Saint-Clair
France

Table of contents

1 Introduction

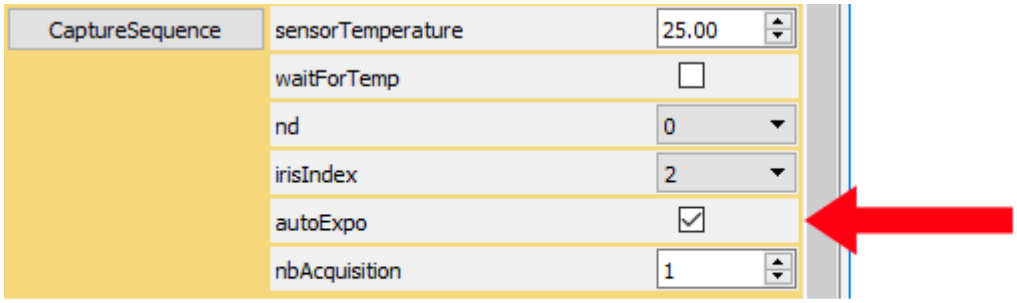
Capture sequence captures necessary images to generate X, Y, Z images.
5 capture are used:
- filter X
- filter Xz
- filter Ya
- filter Yb
- filter Z

2 Exposure time configuration

Capture Sequence exposure time has 3 configurations:
- AutoExposure
- All the capture have the same exposure time
- Exposure time are defined in a json file

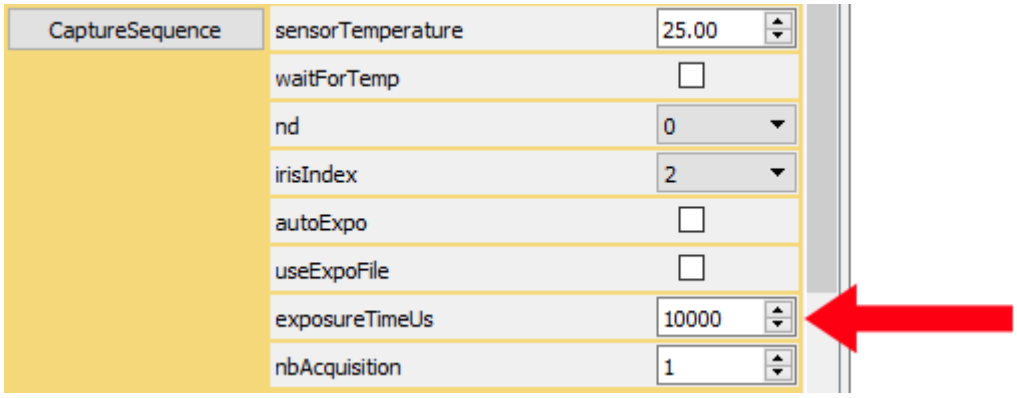
2.1 Auto Exposure

Check "**autoExpo**" so the exposure is calculated during the capture processing



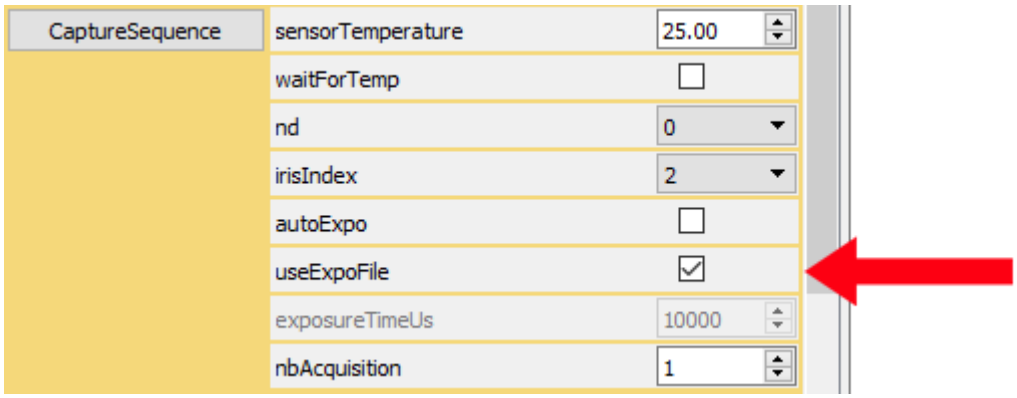
2.2 Manual Exposure

Uncheck "autoExpo" and uncheck "useExpoFile"
Set exposure time in "exposureTimeUs" control.
All the captures are done with this exposure time



2.3 Exposure File

Uncheck “autoExpo” and check “useExpoFile”
File “CaptureSequenceExposureTime.json” is used to retrieve the exposure time



If “CaptureSequenceExposureTime.json” does not exist, it is created with default values.

CaptureSequenceExposureTime.json example:

```
{
   "ExposureTimeUs": {
      "Filter_X": 10001,
      "Filter_Xz": 10002,
      "Filter_Ya": 10003,
      "Filter_Yb": 10004,
      "Filter_Z": 10005
   }
}
```

3 Timing measurement

3.1 Capture Sequence

Following table lists the steps required in the capture sequence:

	step	Step detail	
1	Capture image X	Setup	Moves the filters
		Measure	use AE or not, saved capture or not
		Process	
2	Capture image Xz	Setup	Moves the filters
		Measure	use AE or not, saved capture or not
		Process	
3	Capture image Ya	Setup	Moves the filters
		Measure	use AE or not, saved capture or not
		Process	
4	Capture image Yb	Setup	Moves the filters
		Measure	use AE or not, saved capture or not
		Process	
5	Capture image Z	Setup	Moves the filters
		Measure	use AE or not, saved capture or not
		Process	
6	Generate X, Y, Z		

3.2 Measurement

Measurement depends on the PC performance for **Process** steps.
Setup duration is fixed and constrained by HW

Following figures are done for an exposure time of 1000us. (**Measure** duration will increase with exposure time)

	step	Step detail	No AE	AE meas area: 300x200	AE meas area: full sensor
				Measure is done in several steps of the AE algo	
1	image X	Setup	2.5	2.5	2.5
		Measure	0.5	0.5	1.2
				0.5	1.2
				0.5	1.2
		Process	1.8	2.2	1.8
2	image Xz	Setup	2.5	2.5	2.5
		Measure	0.5	0.5	1.2
				0.5	1.2
				0.5	1.2
		Process	1.8	2.2	1.9
3	image Ya	Setup	2.5	2.5	2.5
		Measure	0.5	0.5	1.2
				0.5	1.2
				0.5	1.2
		Process	1.8	2.1	1.8
4	image Yb	Setup	2.5	2.5	2.5
		Measure	0.5	0.5	1.2
				0.5	1.2
				0.5	1.2
		Process	1.8	2.5	1.9
5	image Z	Setup	2.5	2.5	2.5
		Measure	0.5	0.5	1.2
				0.5	1.2
				0.5	1.2
		Process	1.8	2.2	1.9
6	Generate X, Y, Z				
	Total		**24**	**33**	**40**

All sequences have been done in the same conditions.
AE exposure has been configured to be locked within 2 captures.

4 X, Y, Z images génération

From the 5 pictures captures during the sequence, X, Y, Z are calculated by applying the calibration factor that have been calculated during the calibration.

4.1 Calibration data

From file **OpticalColumn.xml**, ColorCoefComp and ColorCoefCorr are used
Note ColorCoefComp may be replaced by ColorCoef for older version (Unit1)

Calibration data are stored in an array
colorCoefComp[irisIndex][composantType][filterIndex]

where	irisIndex	is the index of the iris mounted on the taprisiot [2, 3, 4, 5]
	composantType	is the type of the image [X, Y, Z]
	filterIndex	is the filter setup [X, Xz, Ya, Yb, Z]

colorCoefCorr[filterIndex][ndIndex]

where	filterIndex	is the filter setup [X, Xz, Ya, Yb, Z]
	NdIndex	is the nd setup [0, 1, 2, 3, 4]

4.2 Processing

irisIndex and NdIndex are the configuration of the Taprisiot

```
comp_X = img_X  * cX  +
         img_Xz * cXz +
         img_Ya * cYa +
         img_Yb * cYb +
         img_Z  * cZ

Where cX  = CoefComp[irisIndex][X][filter_X]  * CoefCorr[filter_X] [NdIndex] / exp(img_X)
      cXz = CoefComp[irisIndex][X][filter_Xz] * CoefCorr[filter_Xz][NdIndex] / exp(img_Xz)
      cYa = CoefComp[irisIndex][X][filter_Ya] * CoefCorr[filter_Ya][NdIndex] / exp(img_Ya)
      cYb = CoefComp[irisIndex][X][filter_Yb] * CoefCorr[filter_Yb][NdIndex] / exp(img_Yb)
      cZ  = CoefComp[irisIndex][X][filter_Z]  * CoefCorr[filter_Z] [NdIndex] / exp(img_Z)


comp_Y = img_X  * cX  +
         img_Xz * cXz +
         img_Ya * cYa +
         img_Yb * cYb +
         img_Z  * cZ

Where cX  = CoefComp[irisIndex][Y][filter_X]  * CoefCorr[filter_X] [NdIndex] / exp(img_X)
      cXz = CoefComp[irisIndex][Y][filter_Xz] * CoefCorr[filter_Xz][NdIndex] / exp(img_Xz)
      cYa = CoefComp[irisIndex][Y][filter_Ya] * CoefCorr[filter_Ya][NdIndex] / exp(img_Ya)
      cYb = CoefComp[irisIndex][Y][filter_Yb] * CoefCorr[filter_Yb][NdIndex] / exp(img_Yb)
      cZ  = CoefComp[irisIndex][Y][filter_Z]  * CoefCorr[filter_Z] [NdIndex] / exp(img_Z)

comp_Z = img_X  * cX  +
         img_Xz * cXz +
         img_Ya * cYa +
         img_Yb * cYb +
         img_Z  * cZ

Where cX  = CoefComp[irisIndex][Z][filter_X]  * CoefCorr[filter_X] [NdIndex] / exp(img_X)
      cXz = CoefComp[irisIndex][Z][filter_Xz] * CoefCorr[filter_Xz][NdIndex] / exp(img_Xz)
      cYa = CoefComp[irisIndex][Z][filter_Ya] * CoefCorr[filter_Ya][NdIndex] / exp(img_Ya)
      cYb = CoefComp[irisIndex][Z][filter_Yb] * CoefCorr[filter_Yb][NdIndex] / exp(img_Yb)
      cZ  = CoefComp[irisIndex][Z][filter_Z]  * CoefCorr[filter_Z] [NdIndex] / exp(img_Z)
```