Flat Field Correction

BRIEF DESCRIPTION
How Flat Field Correction and Contrast Expansion are implemented in AViiVA cameras.

APPLICABLE PRODUCT
AViiVA SM2
AViiVA SC2
AViiVA M4

KEYWORDS
FFC, Flat field correction, Contrast expansion

1. OVERVIEW
By applying per pixel an Ax+B formula, Flat Field Correction (FFC) allows the correction of:

- Dark Signal Non-Uniformity (DSNU),
- Fixed Pattern Noise (FPN)
  (in this text FPN will include these two first parameters),
- CCD Photo-Response Non Uniformity (PRNU),
- Lens vignetting
- Light source non-uniformity
  (in this text PRNU will include these three last parameters).

In AViiVA cameras, the FFC is completed by a Contrast Expansion function.
Note that using digital multiplications with results on integer numbers may cause missing codes.

1.1 ALGORITHM DESCRIPTION.

\[ Y_i = \left( (X_i - b_i) \times (1 + a_i) \right) \times u \times g \]

With:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Y_i )</td>
<td>Corrected pixel value</td>
</tr>
<tr>
<td>( X_i )</td>
<td>Raw pixel value output from the A to D conversion. This value includes the signal, the dark signal and the analog offset set.</td>
</tr>
<tr>
<td>( b_i )</td>
<td>FPN Coefficient: offset correction applied to pixel ( i ) after addition to 1. On 12-bit images, values are computed on 9 bits during calibration procedure.</td>
</tr>
<tr>
<td>( a_i )</td>
<td>PRNU Coefficient: gain correction applied to pixel ( i ) after addition to 1. On 12bit images, values computed on 14 bits, are comprised between 0 and 1. (( a_i = N/16384 ))</td>
</tr>
<tr>
<td>( u )</td>
<td>User programmed offset subtract to each pixel value. It is comprise between –4096 and +4095. This offset will allow contrast expansion and/or noise representation when user wants a &quot;linear&quot; data representation.</td>
</tr>
<tr>
<td>( g )</td>
<td>User programmed gain will be applied to all pixels to perform a contrast expansion. Adjustable value comprised between x1 and x32 (30dB), 8 bits precision : ( g = 1 + N/8 ) with ( N ) : input number 0 to 255.</td>
</tr>
</tbody>
</table>

All calculations are made on at least 14 bits, and then data are output on the needed number of bits (8, 10 or 12).
1.2 SCHEMATIC

**Flat field correction** acts on each individual pixel:
- Step 1 is to correct pixel dark & analog offset signal ($a_i$ is subtracted to each $X_i$ value). Note that calculation will be made with sign to not loose any information.
- Step 2 is to correct pixel gain (each value is multiplied by $b_i$)

**Contrast Expansion** acts on all pixels:
- Step 3 is to add digital offset (offset $u$ is subtracted to each value) If no contrast expansion is used adding a negative value allows to code the noise.
- Step 4 is to use digital gain (each value is multiplied by $g$)
2. FFC CALIBRATION ON AViiVA FAMILLY

AViiVA SM2 (Black & White), AViiVA SC2 (Color) and AViiVA M4 include a Flat Field Correction function. The FFC calibration may be done by three ways:

1. Manual calibration: Available on each product, it allows managing all the parameters for best results. The user can choose the number of acquired lines to be averaged. He can also modify the parameters for some chosen pixels to compensate, for example, a bad calibration calculation due to a bad reference “white paper”.

2. Automatic calibration: Available on AViiVA SM2 and AViiVA SC2, it allows calibrating FFC very easily by sending “calibration” command to the camera. The calculation and average is automatically made inside the camera.

3. Semi-automatic calibration: Available on each product, it consists on acquiring the image on the computer and making the calculations on CommCam software.

Whatever choice is used, it is always possible to modify some factors manually. It is always needed to save the factors after calibration otherwise they will be lost after power off.

Before starting calibrations, the camera must be configured and set up, as it will be in the final application. Any change in the settings (frequency, analog gain, etc…) will require a new calibration.

The “analog offset” values must be superior to zero for good FPN calibration. The default value (128LSB for the 12bit version, 32 for 10bits…) is advised.

3. MANUAL FLAT FIELD CORRECTION

The correction factors must be computed from images taken without FFC and without contrast expansion. To remove noise the acquired lines must be averaged. If not the FFC will introduce a fixed pattern noise.

3.1 FPN DATA.

To acquire these data the camera must be unlighted. For example lens must be closed, covered or light must be turned off.

The number of lines to be averaged depends on the image noise that depends on:

- Gain,
- Integration time,
- CCD temperature.

An average on N line reduce the noise value by a factor $\sqrt{N}$. (N=256 lines gives good results in most of the cases).

For each correction factor the final value must have a lower noise than the one the system is able to detect.

With long integration time and high temperature, CCD temperature changes will require proceeding to a new calibration.

The FPN factor, coded on 9bits for each pixel, is the offset value for a 12-bit output data (LSB of the FPN factor correspond to LSB of the 12-bit data). So when using the camera with only:

- 8 bits output, the value must be multiplied by 16
- 10 bits output, the value must be multiplied by 4

Example:

- Sending 256 will correct an offset of 16 on an 8-bit pixel.
- Sending 342 will correct an offset of 85 on a 10-bit pixel.

3.2 PRNU DATA.

Put a white reference in front of the camera and switch the light on. The quality of the calibration will depend on the reference quality.

To get the best signal to noise ratio, the signal value must be as closed as possible of the camera. Take care to avoid any saturation on the image.

Number of lines to be averaged depends on the camera noise which itself depend on the gain.

The maximum correction factor is x2. The highest pixel correction value must be set to x1 and the others between x1 and x2.

PRNU data format: $<\text{sent value}> = (<\text{calculated gain}> – 1) \times 16384$

The FPN correction must be done first.

Example:

- Sending 0 will apply a gain of 1 to this pixel.
- Sending 8192 will apply a gain of 1.5 to this pixel.
3.3 FFC SERIAL CONTROL.

### 3.4 COMMAND LIST.

<table>
<thead>
<tr>
<th>Command</th>
<th>Range</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write FPN</td>
<td>wfp</td>
<td>Write in the volatile memory specified number of FPN data. If needed a CRC may be used at the end.</td>
</tr>
<tr>
<td>Read FPN</td>
<td>rfp</td>
<td>Read from the volatile memory the specified number of FPN data. CRC ends the return message.</td>
</tr>
<tr>
<td>Write PRNU</td>
<td>wpr</td>
<td>Write in the volatile memory specified number of PRNU data. If needed a CRC may be used at the end.</td>
</tr>
<tr>
<td>Read PRNU</td>
<td>rpr</td>
<td>Read from the volatile memory the specified number of PRNU data. CRC ends the return message.</td>
</tr>
</tbody>
</table>

#### 3.5 EXAMPLE

Commands are composed of:

- **wfp**=<addr> <size> <offset_value> ... <offset_value> [crc16]
- **rfp**=<addr> <size> [with_crc]
- **wpr**=<addr> <size> <gain_value> ... <gain_value> [crc16]
- **rpr**=<addr> <size> [with_crc]

With:

- `<addr>` = decimal address of the first data. (Must be between 1 and the CCD size)
- `<size>` = number of data sent in this command. (Must be between 1 and 5)
- `<offset_value>` = decimal offset value:
  - 0 to 255 on AViiVA SM2 & SC2 (see the note below)
  - 0 to 511 on AViiVA M4
- `<gain_value>` = 0 to 16383 = (pixel gain - 1) x 16384. The pixel gain is x1 to x2.
- `[crc16]` = (facultative) result of a « exclusive or » initialized at 0 on each of the 16 bits.
- `[with_crc]` = flag used or no-used of the CRC:
  - Nothing or 0 = no CRC
  - 1 = with CRC

**wfp=101 5 125 132 140 120 128** will write:

- 125 at address 101
- 132 at address 102
- ...
- 128 at address 105
- No CRC is sent

Sending 8192 PRNU correction data will take about 2mn30.

**Note:** On AviiVA SM2 & SC2, the FPN correction is internally used in 9-bit (0 to 511) but it is decomposed into a pixel offset value (read or write as described above) and common odd & even offset values that are accessible at:

- `<addr>` = CCD size + 1 (for common odd offset value)
- CCD size + 2 (for common even offset value)
4. **« AUTOMATIC » FLAT FIELD CORRECTION.**

The AViiVA SM2 and AViiVA SC2 cameras are able to calculate internally the correction factors. Mean lines in darkness and under light are computed on 20bits.

For each of FPN and PRNU calibration processes, the camera acquires an image of **256 lines**.

The Camera has to be ready for the grab (or at least waiting for the trig signal) before sending the calibration command.

4.1 **FPN CALIBRATION**

1- Put the camera in darkness.

2- Start FPN automatic calibration by sending "!=5" command or using CommCam software.

   ➢ The camera resend a return message (see bellow)

   ➢ At this state, the FPN factors are stored in a volatile memory.

3- Store the factors in the chosen bank

4.2 **PRNU CALIBRATION**

1- Replace the scene by a white reference. Be careful: the calibration quality will depend on the reference quality.

2- Put the light on and set all the parameters as in the final application (light, lens, gain, etc…).

   ➢ The white level should be in the second half of the dynamic (between 2048 and 4095 in 12-bit).
   ➢ As the pixels gain value is calculated between x1 and x2, the pixels with lower value will be clamped with x2 gain.
   ➢ None pixel can be saturated
   ➢ The best result will be obtained when pixels values are just under the maximum.

4- Start PRNU automatic calibration by sending "!=6" command or using CommCam software.

   ➢ The calibration will take some seconds. It can be aborted by "!=9" command.
   ➢ The camera resend a « return » message (see bellow)

   ➢ At this state, the FPN factors are stored in a volatile memory.

5- Storage the factors by choosing one of the bank

4.3 **RETURNED MESSAGES AND STATUS READOUT**

4.3.1 **RETURNED MESSAGES**

Each command sent to the camera is followed by a status sent back by the camera: “\textasciitilde ret\_code”

<table>
<thead>
<tr>
<th>Possible values for “ret_code”</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>Parameter out of range</td>
</tr>
<tr>
<td>2</td>
<td>Protocol error</td>
</tr>
<tr>
<td>4</td>
<td>Invalid command</td>
</tr>
<tr>
<td>5</td>
<td>Transmission error</td>
</tr>
<tr>
<td>6</td>
<td>Hardware access error</td>
</tr>
<tr>
<td>7</td>
<td>No privilege</td>
</tr>
<tr>
<td>8</td>
<td>Power-on error</td>
</tr>
</tbody>
</table>

4.3.2 **STATUS READOUT**

After each calibration command, the user can get informations by a status readout command: “!=4”.

Here is the status register format:

<table>
<thead>
<tr>
<th>B15-12</th>
<th>B11</th>
<th>B10</th>
<th>B9</th>
<th>B8</th>
<th>B7-3</th>
<th>B2</th>
<th>B1</th>
<th>B0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>Cal End</td>
<td>Uvf</td>
<td>Ovf</td>
<td>Div0</td>
<td>00000</td>
<td>Ext clock</td>
<td>Trig2</td>
<td>Trig1</td>
</tr>
</tbody>
</table>

Trig1, Trig2, Ext clock : if =1 : the trigger signal is received by the camera (used for debug)

Div0 : if = 1 : internal calculation error. The calibration will not be achieved

Ovf : if = 1 : Warning = Over Illumination during calibration.

Uvf : if = 1 : Warning = Under Illumination during calibration
Cal end : if = 1 : The calibration is finished.
In case of warning, the calibration is made but some coefficients may be bad or clamped.
When using "commcam" configuration software, the status is automatically read and clearly displayed after calibration.

5. « SEMI-AUTOMATIC » FLAT FIELD CORRECTION.
The CommCam software may be use to compute the correction factor from specified raw images.
Select the Dark and Light images previously saved
Built the file : Commcam will compute the data and built a file in the good format
Send this file to the camera
It is also possible to read the data from the camera, to save it into a file, to modify it and resend it to the camera.
The file is in a txt format with :
- 1st row : number of the pixel
- 2nd row : PRNU coefficient
- 3rd row : FPN coefficient
Each row is separated by a "tab"

6. DATA STORAGE.
The AVIIVA cameras will allow:
- 4 storage "banks" to store the PRNU data
- 4 storage "banks" to store the FPN data.
The user will have to choose which of these 8 banks he want to use. At power on the camera will automatically use the 2 most recently used.
All these banks are empty when the camera is delivered.
The serial line may be used to write or read the volatile memory content. Special order may be used to store the data in non-volatile memories.