FLIR Systems
Advanced Thermal Solutions

DL002U-E
ALTAIR Refence Guide

ALTAIR

REFERENCE GUIDE

CONTENTS

1. LIST OF FILE TYPES

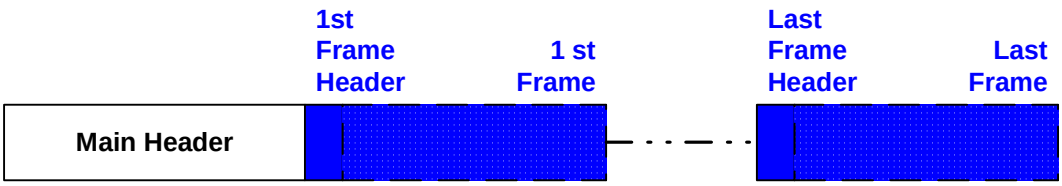
Extension	File Type	Data Unit	Viewer
PTW	**IR** image or film	DL	Altaïr
PTM	**Average** IR single image	DL	
PTB	**Noise** IR single image	~DL	
PTS	Mathematical Calculation image or film	DL/~DL	
PTR	PTR DOS 1 IR image or film	DL	
PT2	PTR DOS 2 IR image or film	DL	
PIX	BPR (Bad Pixel Replacement) File	DL	BPR Viewer
COE	NUC (Non Uniformity Correction) File	DL	NUC Viewer
MAP	SCD Remapping File	n/a	
EXP	Temperature Calibration File	DL vs °C	Calibration File Manager
EX2	Temperature Calibration File for temperature Correction	DL vs °C	
TRI	Lut file for Tri board		
NUC	Nuc Information File		
CALLISTO Characterization & Calibration			
PTD	IR Detectivity image (D*)	$cm.Hz^{1/2}.W^{-1}$	Altaïr
PTK	NETD IR Image	mK	
PTT	Sensitivity Image	mK/DL	
SATIR Multiple Sensor Signature Analysis			
PT0	Near **IR** Image	DL	Altaïr
PT1	**IR** Image of film **SWIR** waveband	DL	
PT2	**IR** Image or film **MWIR** waveband	DL	
PT3	**IR** Image or film **LWIR** waveband	DL	
PTV	**Visible** image or film	DL	
Altaïr LI Lockin			
PTP	Phase image	°	Altaïr-LI / Altaïr
PTW	Module image	DL	
PTA	A image for motion compensation	DL	
PTB	B image for motion compensation	DL	
PTR	Result for Random or Motion Compensation	DL	
ORION Rotating Filter Camera			
PT1	**1st** filter IR Image of film	DL	Altaïr
PT2	**2nd** filter IR Image of film	DL	
PT3	**3rd** filter IR Image of film	DL	
PT4	**4th** filter IR Image of film	DL	
PT5	**5th** filter IR Image of film	DL	
PT6	**6th** filter IR Image of film	DL	

2. BASIC DATABASE TYPES

Data Type	Size (bits)	Min Range	Max Range
unsigned char	8	0	255
char	8	-128	127
enum	16	-32,768	32,767
unsigned int	32	0	4,294,967,295
short int	16	-32,768	32,767
int	32	-2,147,483,648	2,147,483,647
unsigned long	32	0	4,294,967,295
long	32	-2,147,483,648	2,147,483,647
float	32	$3.4 * (10e^{-38})$	$3.4 * (10e^{+38})$
double	64	$1.7 * (10e^{-308})$	$1.7 * (10e^{+308})$
long double	80	$3.4 * (10e^{-4932})$	$1.1 * (10e^{+4932})$

3. IR IMAGE FILE FORMAT

3.1. Database Structure



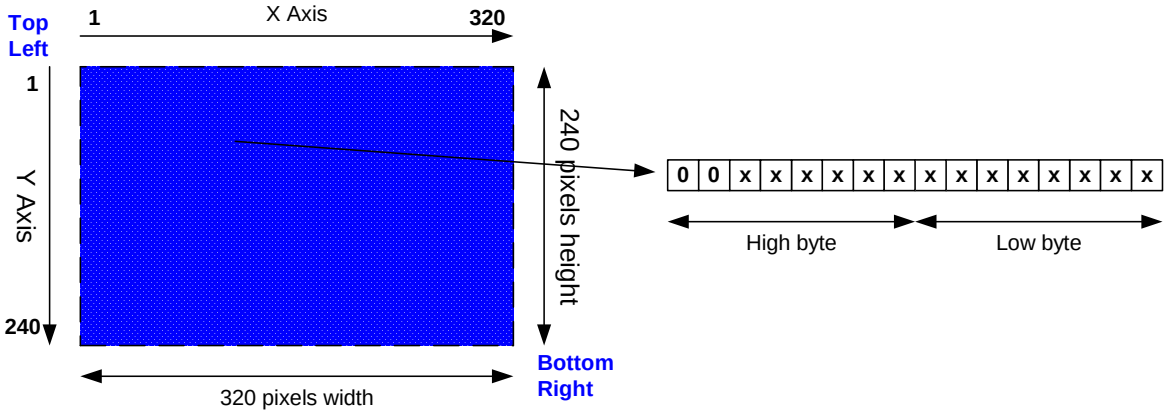
Inherited from the FLIR SYSTEMS PTR proprietary file format, the IR database is a binary file according to the above structure.

This architecture is based on a **main header** describing the database itself; all parameters of the file itself are self-contained; additionnaly it includes all the parameters of acquisition as:

- size of main header,
- size of frame's header
- number of frames acquired
- Image format
- Date / time
- A/D digitization depth
- Camera information
- Etc.

Size of this main header is generaly a multiple of 1 KByte (1024 bytes). In anycase this size is expressed in the very beginning of the header itself allowing a 2 pass dynamic file reading by 1st reading sizes, allocating buffers consequently then reading in the 2nd pass all the data.

Following this main header is the stream of frames. Each frame includes its own header mainly for time stamp informations. Then we get pixels of the frame according to the size of the array. For instance for a 320*240 array, we get 76800 pixels consecutively. 1st pixel is top left (x:1;y:1), the 2nd one is (x:2;y:1),…, the last one is bottom right (x:320;y:240).



A pixel is digitised up to 16 bits, in case of lower A/D conversation the upper bits are set to 0.

3.2. Main header format

Used structures:
PTR_TIME:

Data Type	Designation & Size	Description	Offset (bytes)	Version Example*
byte	Minute	Minute	**0**	13
byte	Heure	Hour	**1**	10
byte	Centieme	Cent	**2**	56
byte	Seconde	Second	**3**	24

PTR_DATE:

Data Type	Designation & Size	Description	Offset (bytes)	Version Example*
WORD	Annee	Year	**0**	2004
char	Jour	Day of the month	**2**	17
char	Mois	Month (1 = January)	**3**	02

PTR_LOCKIN_DATA:

Data Type	Designation & Size	Description	Offset (bytes)	Version Example*
WORD	wLockinBoxVersion	Version of Lockin Process	**0**	3
DWORD	dwLockinPeriod	Period of signal	**2**	102440
DWORD	dwLockinPhase	Phase of signal	**6**	32145
bool	bLockinMin	Minimum of signal ?	**10**	FALSE
bool	bLockinMax	Maximum of signal ?	**11**	FALSE
BYTE	bySignalNumber	Nb signals in next buffer	**12**	4
WORD	wLockinMeanValue [4]	Buffer of signals value	**13**	8192

PTR_HEADER:

Data Type	Designation & Size	Description	Offset (bytes)	Version Example*
char	Signature[5]	Signature	**0**	PTR
char	Version[5]	Version number	**5**	
char	FinDeFichier	End of file ASCII code	**10**	
DWORD	TailleHeaderFilm	Main header size (**bytes**)	**11**	
DWORD	TailleHeaderTrame	Frame header size (**bytes**)	**15**	
DWORD	TailleBloc	Size of 1 frame + its header	**19**	
DWORD	TailleTrame	Size of 1 frame	**23**	
DWORD	NombreTrame	Number of field in the file	**27**	
DWORD	NumeroTrame	Current field number	**31**	
PTR_DATE	Date	File save date	**35**	
PTR_TIME	Time	File save time	**39**	
unsigned char	Millieme		**43**	
char	Camera[20]	Camera name	**44**	Jade LR
char	Lens[20]	Lens name	**64**	
char	Filter[20]	Filter name	**84**	
char	Aperture[20]	Aperture name	**104**	
float	fBilletSpeed	Billet speed in m/s	**124**	***IRUS***
float	fBilletDiameter	Billet diameter in m	**128**	

WORD	wBilletShape	Billet shape (squared:0 1:round)	**132**	
char	Reserve0[7]	Reserved	**134**	
float	fEmissivity	Emissivity (<= 1)	**141**	
float	fAmbiant	Background temperature (in K)	**145**	
float	fDistance	Distance (in meters)	**149**	
byte	bCoil	Inductor coil 0:unknown 1:small 2:big	**153**	***IRUS***
long	lPower	Inductor power in W	**154**	
WORD	wVoltage	Inductor voltage in V	**158**	
long	lFrequency	Inductor Frequency in Hz	**160**	
char	Reserved[5]	Reserved	**164**	
byte	bSynchronization	Synchronization	**169**	
float	Transmission	Atmospheric Transmission (%)	**170**	
float	Extinction	Extinction coefficient	**174**	
WORD	bObject		**178**	
WORD	bOptic		**180**	
WORD	bAtmo		**182**	
float	Atmosphere	Atsmosphere Temperature (K)	**184**	
float	CutOn	Cut On wavelength @ 50% (µm)	**188**	
float	CutOff	Cut Off wavelength @ 50 % (µm)	**192**	
float	fPixelSize	Pixel size in µm	**196**	
float	fPixelPitch	Pixel pitch in µm	**200**	
float	fAperture	(F/#) Detector aperture	**204**	
float	fFocal	Optics focal length (mm)	**208**	
float	fHousing	Housing internal temperature	**212**	
float	fHousing2	2nd Housing internal temperature	**216**	
char	SerialNumber[11]	Camera serial number	**220**	
char	Reserve1[14]	Reserved	**231**	
WORD	MinLUT	Minimum level threshold	**245**	
WORD	MaxLUT	Maximum level threshold	**247**	
char	Reserve2[28]	Reserved	**249**	
WORD	EchelleSpeciale	(0 : Regular scale ; 1 : Special scale)	**277**	
char	UniteEchelle[10]	Unit for special scale	**279**	
float	Echelle[16 + 1]	Values for special scale	**289**	
float	fGain	Gain scale factor for lockin images	**357**	
float	fOffset	Offset scale factor for lockin images	**361**	
float	fZoomX	Horizontal zoom factor	**365**	
float	fZoomY	Vertical zoom factor	**369**	
char	Reserve3[2]	Reserved	**373**	
char	Reserve4[2]	Reserved	**375**	
WORD	NombreColonne	Pixels per line	**377**	
WORD	NombreLigne	Lines per field	**379**	
WORD	NombreBit	A/D Dynamic	**381**	
WORD	Moyenne	Temporal frame average depth	**383**	***SATIR / SATURNE***
float	fLongitude	Longitude Location in °	**385**	
float	fLatitude	Latitude Location in ° (- means South)	**389**	

float	fAltitude	Altitude location in m	**393**	
byte	bExternalSynchro	TRUE:External,FALSE:Internal	**397**	
char	Reserve5[5]	Reserved	**398**	
float	fPeriod	Acquisition period (in s)	**403**	***FLIR***
float	fIntegration	Integration time (in s)	**407**	***FLIR***
WORD	bWindow	Subwindowing capability	**411**	***WOLF***
float	fIntergrationSPI [6]	Integration times (in s)	**413**	***ORION***
char	szFilterSPI [6][20]	Filter names	**437**	
WORD	wNuc	Nuc table	**557**	
char	Reserve6 [4]	Reserved	**559**	***FLIR***
char	Parametre [1000]	Comment	**563**	
char	Etalonnage [100]	Calibration file name	**1563**	
char	ToolsFile [256]	Tools file name for Altair	**1663**	
byte	bIndexPaletteValid	Is index of palette valid?	**1919**	
WORD	wIndexCurrentPalette	Index of current palette	**1920**	
byte	bTogglePalette	Palette toggle?	**1922**	
byte	bPaletteAgc	Palette AGC?	**1923**	
byte	bIndexUnitValid	Is index of unit valid?	**1924**	
WORD	wIndexCurrentUnit	index of current unit (DL / Temp / Radiance / Percent / Special ...)	**1925**	
POINT	PointScrollPos	Position of zoom (unit scale 1:1)	**1927**	
byte	bKeyFrameNumber	Key frames number	**1935**	
DWORD	dwKeyFrames[30]	Key frames in film	**1936**	
byte	bLockedToKeyFrame	Is player locked to keyframes ?	**2056**	
byte	dwROIValid	Frame selection [start;end] valid	**2057**	
DWORD	dwROIStart	Frame number of ROI start	**2058**	
DWORD	dwROIEnd	Frame number of ROI end	**2062**	
byte	bLockedToROI	Is player locked to ROI?	**2066**	
byte	bLoopPlayer	Is player running in infinite loop?	**2067**	
DWORD	dwInitPlayerFrame	Init player frame	**2068**	
byte	byIsothermActive0	Palette : is isotherm 0 active ?	**2072**	
WORD	wIsothermDLMin0	Palette : isotherm 0 DL min	**2073**	
WORD	wIsothermDLMax0	Palette : isotherm 0 DL max	**2075**	
COLORREF	colorIsothermColor0	Palette : isotherm 0 color	**2077**	
byte	byIsothermActive1	Palette : is isotherm 1 active ?	**2081**	
WORD	wIsothermDLMin1	Palette : isotherm 1 DL min	**2082**	
WORD	wIsothermDLMax1	Palette : isotherm 1 DL max	**2084**	
COLORREF	colorIsothermColor1	Palette : isotherm 1 color	**2086**	
byte	byIsothermActive2	Palette : is isotherm 2 active ?	**2090**	
WORD	wIsothermDLMin2	Palette : isotherm 2 DL min	**2091**	
WORD	wIsothermDLMax2	Palette : isotherm 2 DL max	**2093**	
COLORREF	colorIsothermColor2	Palette : isotherm 2 color	**2095**	
byte	byZeroActive	Palette : is zero active ?	**2099**	
WORD	wZeroDL	Palette : zero DL	**2100**	
WORD	wPaletteWidth	Palette : width of logpalette display	**2102**	
byte	byPaletteFullScaleDisplay	Palette : is displaying full scale ?	**2104**	
byte	byPtrFrameType	PTR frame buffer type	**2105**	

		(0==WORD 1==double ...)		
double	dKm	Thermoelasticity coefficient	**2106**	
float	fDemodeFrequency	Demodulation frequency for this film	**2114**	
DWORD	dwAxeTypeId	Cordinate system : Axe type	**2118**	
int	nOriginX	Cordinate system : origin X	**2122**	
int	nOriginY	Cordinate system : origin Y	**2126**	
BYTE	byShowOrigin	Cordinate system : show origin	**2130**	
COLORREF	colorAxe	Axe color	**2131**	
int	nAxeSize	Axe size	**2135**	
BYTE	bAxeValid	system axe is valid	**2139**	
float	fDistanceOffset	distance offset (m)	**2140**	
BYTE	bHistoEqualisationEnable	histo equalisation enable	**2144**	
WORD	wHistoEqualisationPercent	histo equalisation percent [0 - 100]	**2145**	
char	Etalonnage2[256]	Calibration file name	**2147**	
bool	bDriverTopFrameValid	qwRealDriverTopFrame of PTR_IMAGE is valid	**2403**	
int	nSubSampling	record subsampling	**2404**	
BYTE	bCaptureHorizontalFlip	Camera Horizontal Flip	**2408**	
BYTE	bCaptureVerticalFlip	Camera Vertical Flip	**2409**	
float	fCaptureBlackBodyTemp	Black Body Temp. (K)	**2410**	
BYTE	byCaptureWheelIndex	Capture Wheel index	**2414**	
BYTE	byCaptureFocalIndex	Capture Focal index	**2415**	
BYTE	byCaptureFlipValid	CNUC Flip Valid	**2416**	
char	szLendRef[20]	Lens reference	**2417**	
char	szReserved7 [591]	Reserved	**2437**	
DWORD	dwSizeByte	Hypercal structure size	**3028**	
DWORD	dwCalibrationTypeId	Hypercal calibration type	**3032**	
DWORD	dwLinearDLMin	Minimum Valid DL	**3036**	
DWORD	dwLilneraDLMax	Maximum Valid DL	**3040**	
float	fCalibTempMin	Mini temperature range	**3044**	
float	fCalibTempMax	Max temperature range	**3048**	
float	fOffsetDL		**3052**	
float	fOffsetIT		**3056**	
float	fTransmissionOpt	Hypercal Param	**3060**	
float	fStreamOffset		**3064**	
float	lLogSlope		**3068**	
float	fLogIntercept		**3072**	
BYTE	byFramatome[400]	Special Framatome	**3076**	

Example is given for a file time of **23h 59' 01'' 456ms** & a date of the **19th of December 2002**.

3.3. Frame header format

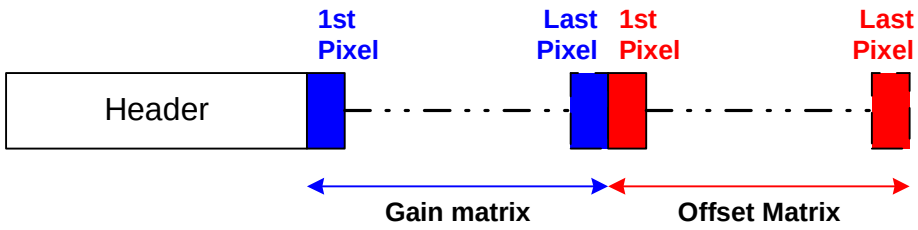
PTR_IMAGE:

Data Type	Designation & Size	Description	Offset (bytes)	Example*
char	Reserve0[80]	Reserved	**0**	
PTR_TIME	Time	Frame time	**80**	59
char	Reserve1[76]	Reserved	**84**	
byte	Millieme	Frame Thousands	**160**	6
WORD	Millionieme	Frame Millions	**161**	789
float	fDemodeFrequency	Demodulaton frequency for this image	**163**	
char	Reserve2[61]	Reserved	**167**	
float	fDetectorTemp	detector temperature (K) (0==disable)	**228**	
float	fSensorTemp[4]	housing temp (K) (0==disable)	**232**	
WORD	wLockinBoxVersion	Lockin Version (0==no lockin data)	**248**	
DWORD	dwLockinPeriod	Lockin period (seconde)	**250**	
DWORD	dwLockinPhase	Lockin Phase (?)	**254**	
bool	bLockinMin	Lockin Minimum	**258**	
bool	bLockinMax	Lockin Maximum	**259**	
BYTE	bySignalNumber	Number of signals	**260**	
WORD	wLockinMeanValue[4]	4 signals mean values	**261**	
DWORD	dwLockinSpare[2]	lockin spare	**269**	
byte	byMultiItNumber	MultiIt number	**277**	
WORD	bWnd	Sub-windowing 1:On ; O:Off	**278**	
int16	nWndLeft	Left coordinate	**280**	
int16	nWndTop	Top coordinate	**282**	
int16	nWndWidth	Window width	**284**	
int16	nWndHeight	Window height	**286**	
float	fIntegration	Integration Time (µs)	**288**	
QWORD	qwRealDriverTopFrame	real top frame counter of driver	**292**	
byte	byPageIndex	Page index for BF1 detector	**300**	
QWORD	qwCameraTimeStamp	Time stamp from camera (µs)	**301**	
char	szReserved3[707]	Reserved	**309**	

Example is given for a frame time of **23h 59’ 01’’ 456ms 789 µs**.

4. NUC (NON UNIFORMITY CORRECTION)

4.1. Database Structure



4.2. Database File Format

Data Type	Designation & Size	Description	Offset (bytes)
char	Signature[5]	String "**COE**"	**0**
char	Version[5]	Version number	**5**
char	EndOfFile	Reserved PTR	**10**
char	Reserve[181]	Reserved	**11**
char	Camera[20]	Camera name	**192**
char	CameraSerial[11]	Camera serial number	**212**
char	FilterName[20]	Name of the filter the NUC was made with	**223**
float	Integration	Integration time the NUC was made with	**243**
byte	bWndMode	Windowing mode the NUC was made with	**247**
WORD	wLeft	Left coordinate of the ROI	**248**
WORD	wTop	Top coordinate of the ROI	**250**
WORD	NombreColonne	Number of pixels/line	**252**
WORD	NombreLigne	Number of lines/frame	**254**
long	**Size**	Number of pixels in the array	**256**
float	Alpha[**Size**]	Alpha Gain table (**Size** elements)	**260**
float	Beta[**Size**]	Beta Offset table (**Size** elements)	**260+4*Size**

Warning: For compatibility reasons with former file format, when the fields **NombreColonne** and **NombreLigne** return 0, their value should be replaced by **128**.

4.3. Gain & Offset formula

α and β coefficients are applied on image to correct FPA non uniformity. 2 point correction allows applying one gain and offset independent for each pixel of the image.

To compute these 2 images, 2 reference images at 2 different temperatures are needed. These images will be acquired on a uniform black body.

A first image **I1** is saved at homogeneous temperature **T1**.
A second image **I2** is saved at homogeneous temperature **T2**.
We define FPA pixel number by **N**.

We compute spatial average **M1** for each pixel of the first image:

$$M_1 = \frac{\sum_{i=1}^{N} I_{1_i}}{N}$$

We compute spatial average **M2** for each pixel of the second image:

$$M_2 = \frac{\sum_{i=1}^{N} I_{2_i}}{N}$$

For each pixel of the image, we compute an image of gain (α) with the following formula :

$$\alpha_i = \frac{M_2 - M_1}{I_{2_i} - I_{1_i}}$$

For each pixel of the image, we compute an image of offset (β□ with the following formula :
For computing offset array 2 methods can be applied:

- mean value method with the following formula :

$$\beta_i = M_1 - \alpha_i * I_{1_i}$$

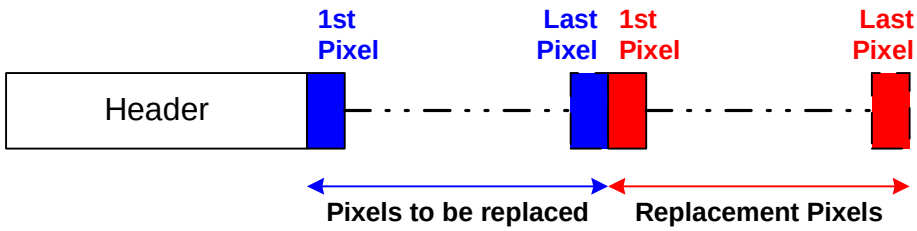
- zero method with the following formula (with I = the coldest image image between I1 and I2) :

$$\beta_i = -\alpha_i * I_i$$

This coefficients α and β are images and take different value for each pixel of the image.

5. BPR (BAD PIXEL REPLACEMENT)

5.1. Database Structure



5.2. Database File Format

Data Type	Designation & Size	Description	Offset (bytes)
char	Signature[5]	String "**PIX**"	**0**
char	Version[5]	Version number	**5**
char	EndOfFile	Reserved PTR	**10**
char	Reserve[237];	Reserved	**11**
int	Interpolation;	Interpolated file (obsolet)	**248**
char	Reserve[2];	Reserved	**250**
int	NombreColonne	Number of pixels/line	**252**
int	NombreLigne	Number of lines/frame	**256**
int	**Size**	Number of bad pixel	**260**
long	Defectueux[**Size**]	List of Pixels to be replaced (**Size** elements)	**264**
long	Remplacant[**Size**]	List of replacement Pixels (**Size** elements)	**264+Size**

Warning: For compatibility reasons with former file format, if fields **NombreColonne** and **NombreLigne** return 0, their value should be **128**.

5.3. Replacement Algorithm

One bad pixel will be replaced by one of its neighboor itself not bad. Replacement algorithm is looking in the 48 neighboor pixels; the sort engine searching is presented in the following table:

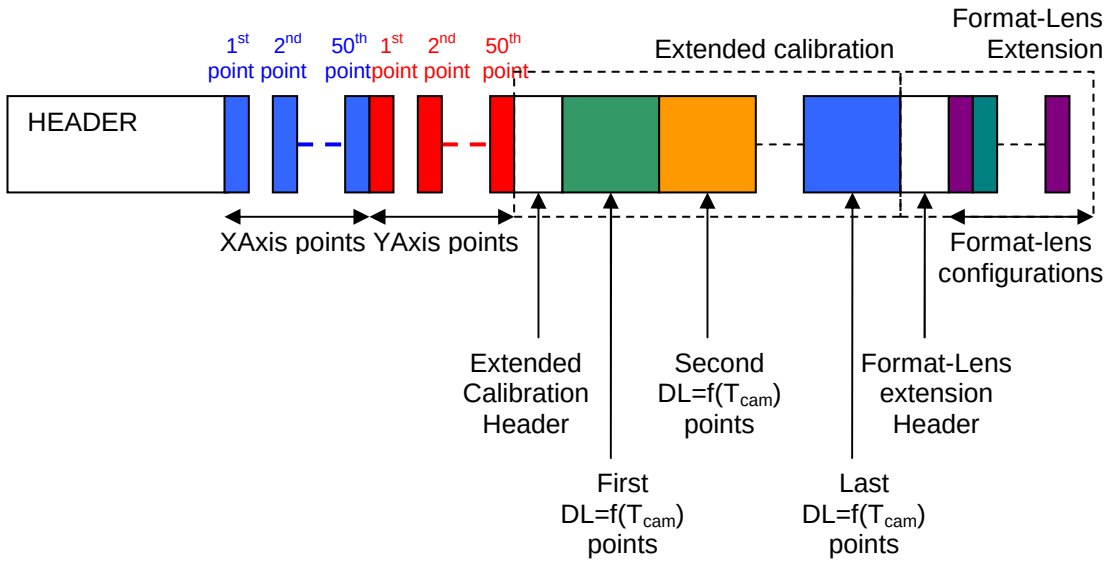
Numbers in cell are sorted numbers. The bad pixel to replace is located at the center of the table (x:5,y:5). In the case of one pixel near a border or a corner, non existing cells are not taking into account for replacement.

Y \ X	0	1	2	3	4	5	6
0	45	37	29	25	30	38	46
1	44	21	13	9	14	22	39
2	36	20	5	1	6	15	31
3	28	12	4	☒	2	10	26
4	35	19	8	3	7	16	32
5	43	24	18	11	17	23	40
6	48	42	34	27	33	41	47

In the case of an alone pixel (3, 3), it will be replaced by its above pixel (3,2). If this replacement pixel is also bad, its right neighboor (4,3) will be accepted , etc...

6. TEMPERATURE CALIBRATION

6.1. Database Structure



6.2. Database File Format

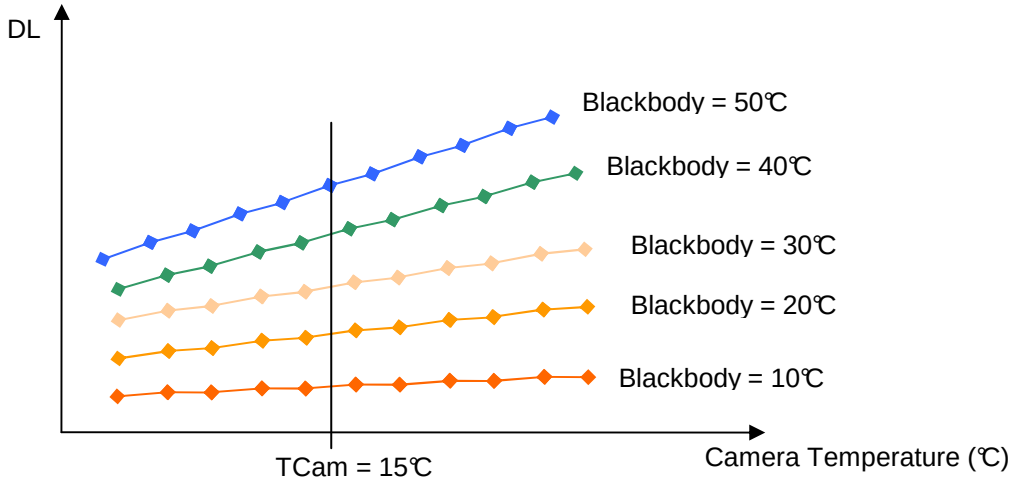
Data Type	Designation & Size	Description	Offset (bytes)
char	szSignature [5]	String « **EXP** »	**0**
char	szVersion[5]	Version Number	**5**
char	szCamera[20]	Camera Name	**10**
char	FinDeFichier	Reserved	**30**
WORD	wIntegration	Integration Time (**µs**)	**31**
WORD	wFrameRate	Frame Rate (**Hz**)	**33**
float	fAperture	F/# Aperture	**35**
float	fBackground	Background temperature in °C	**39**
float	fHousingTemp	Camera Housing Temp in K	**43**
byte	bExtendedCalibration	Is Extended calibration enable	**47**
char	Reserve1 [89]	Reserved	**48**
struct date	Date	Calibration Date	**137**
char	szLens [10]	Lens Name	**141**
char	szFilter [20]	Filter Name	**151**
float	fDistance	Calibration Distance (**m**)	**171**
char	szUniteX [10]	X axis unit	**175**
char	szUniteY [10]	Y axis unit	**185**
char	szLabelX [20]	X axis label	**195**
char	szLabelY [20]	Y axis label	**215**
char	Reserve3[30]	Reserved	**235**
WORD	**wTaille**	Amount of points in the following curve (**<=50**)	**265**
float	fX[50]	Up to 50 X axis points (**wTaille** useful elements)	**267**
float	fY[50]	Up to 50 Y axis points (**wTaille** useful elements)	**467**
If bExtendedCalibration & 0x01= 0x01			

long	lNbBBTemp	Number of blackbody temperature	**667**
char	Reserved[60]	Reserved	**671**
float	fBBTemp	Blackbody temperature	**731**
long	lNbValues	Nb of TCam values for this BBTemp	**735**
float	fTCam[lNbValues]	Camera temperature values	**739**
float	fDL[lNbValues]	DLs corresponding to camera temperatures	
float	fBBTemp2	Blackbody temperature	
long	lNbValues2	Nb of TCam values for this BBTemp	
float	fTCam[lNbValues2]	Camera temperature values	
float	fDL[lNbValues2]	DLs corresponding to camera temperatures	
… until n == lNBBBTemp-1			
float	$fBBTemp_n$	Blackbody temperature	
long	$lNbValues_n$	Nb of TCam values for this BBTemp	
float	fTCam[n]	Camera temperature values	
float	fDL[n]	DLs corresponding to camera temperatures	
If bExtendedCalibration & 0x02 = 0x02			
long	lNBConfig	Number of additional format-lens configuration	
char	Reserved[60]	reserved	
char	LensName[10]	Lens name	
__int16	nLeft	ROI(format)'s left coordinate	
__int16	nTop	ROI(format)'s top coordinate	
__int16	nWidth	ROI(format)'s width	
__int16	nHeight	ROI(format)'s height	
float	fAlpha	Gain of Configuration's affine transformation	
float	fBeta	Offset of Configuration's affine transformation	
char	LensName[10]	Lens name	
__int16	nLeft	ROI's left coordinate	
__int16	nTop	ROI's top coordinate	
__int16	nWidth	ROI's width	
__int16	nHeight	ROI's height	
float	fAlpha	Gain of Configuration's affine transformation	
float	fBeta	Offset of Configuration's affine transformation	
… until n == lNBConfig-1			
char	LensName[10]	Lens name	
__int16	nLeft	ROI's left coordinate	
__int16	nTop	ROI's top coordinate	
__int16	nWidth	ROI's width	
__int16	nHeight	ROI's height	
float	fAlpha	Gain of Configuration's affine transformation	
float	fBeta	Offset of Configuration's affine transformation	

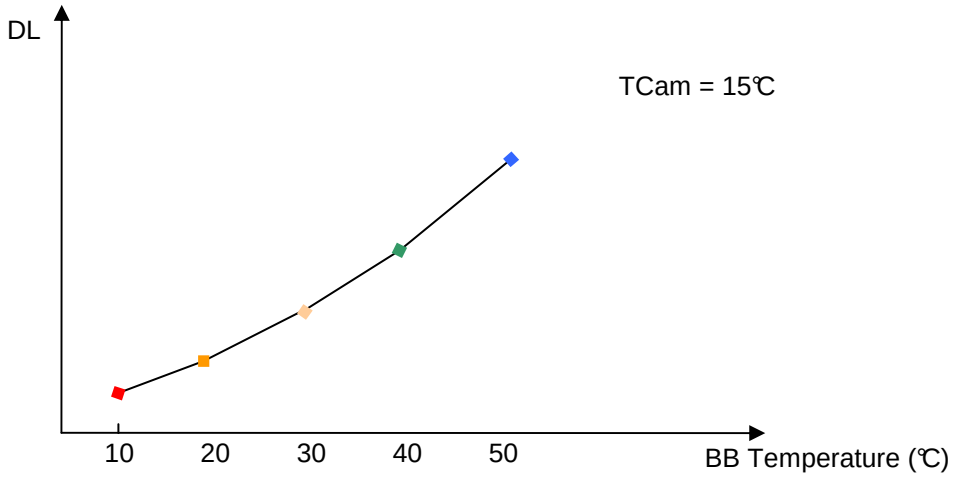
6.3. Temperature Correction

In order to compensate temperature drift due to the internal camera housing effect on the measurement, the following method is used for correction:

During calibration process, the response of the camera is acquired for many camera housing temperatures, and for all the temperature calibration points needed. This gives the reponses of the camera over housing temperature.



The calibration curve for a specific housing temperature is interpoled from this curves beam. For each blackbody temperature, the TCam points that surround the needed housing temperature are used to interpolate the corresponding DL value. This gives a temperature compensated curve:



6.4. Additional format-lens configurations

It has been proven that the relation between the digital level DL_A measured with a format-lens configuration A and the digital level DL_B measured with a format-lens configuration B for the same balackbody and camera temperature is a n affine transformation. Thus we have:

$$DL_B = \alpha\, DL_A + \beta$$

These coefficients are kept in each additional format lens configuration and allow to build the configuration's calibration curves by applying the transformation to the main configuration's curve.

6.5. Multiple Integration Time Naming convention

Images acquired in multiple integration time mode are saved using the following naming convention:
SelectedFilename_X where X correspond to the integration time index.

7. TRI

7.1. Database File Format

Data Type	Designation & Size	Description	Offset (bytes)
char	Signature[5]	String « **TRI** »	**0**
char	Version[5]	Version Number	**5**
char	EndOfFile	Reserved PTR	**10**
char	Reserve[241]	Reserved	**11**
long	NbValue	Nb Value in LUT	**252**
char	NucFilePath[260]	Path to Nuc File	**256**
char	Calibration1 [260]	Path to First Calibration File	**516**
char	Calibration2 [260]	Path to second Calibration File	**776**
char	Calibration3 [260]	Path to third Calibration File	**1036**
float	Threshold1	1st Threshold	**1296**
float	Threshold2	2nd Threshold	**1300**

8. CALCULATION FORMULA

8.1. Temperature Calculation Formula

$$Q = \tau_{atm} * [\varepsilon * f(T_{obj}) + (1- \varepsilon) * f(T_{bkg})] + (1 - \tau_{atm}) * f(T_{atm})$$

where Q is quantity of radiation (**DL**)

ε is emissivity of the object

T_{obj} is temperature of the object (**K**)

T_{bkg} is background temperature (**K**)

T_{atm} is temperature of the atmosphere (**K**)

τ_{atm} is atmosphere transmission factor (**%**)

f is the calibration function

8.2. Radiance Calculation Formula

$$L = \int_{\lambda CutOn}^{\lambda CutOff} Planck(\lambda, T)$$

where L is radiance (**w/m²/sr**)

λ is wavelength (**µm**)

λ_{CutOn} is Cut On wavelength (**µm**)

λ_{CutOff} is Cut Off wavelength (**µm**)

T is temperature of the object (**K**)

Planck is Planck's law

8.3. Transmission Calculation Formula

$$\tau_{atm} = e^{(-d * a)}$$

where τ_{atm} is transmission factor

d is distance of the object (km)

a is extinction coefficient (km^{-1})

8.4. Rms noise and area standard deviation formula

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} [X_i - \overline{X}]^2}{N}}$$

where **N** is the amount of images within the film