Ambarella

SDK6 API Exif

Draft Version 0.2

December 23, 2014

Ambarella

Confidentiality Notice:

The contents of this document are proprietary and confidential information of Ambarella Inc.

The material in this document is for information only. Ambarella assumes no responsibility for errors or omissions and reserves the right to change, without notice, product specifications, operating characteristics, packaging, ordering, etc. Ambarella assumes no liability for damage resulting from the use of information contained in this document. All brands, product names, and company names are trademarks of their respective owners.

US
3101 Jay Street
Ste. 110
Santa Clara, CA 95054, USA
Phone: +1.408.734.8888
Fax: +1.408.734.0788

Hong Kong
Unit A&B, 18/F, Spectrum Tower
53 Hung To Road
Kwun Tong, Kowloon
Phone: +85.2.2806.8711
Fax: +85.2.2806.8722

Korea
6 Floor, Hanwon-Bldg.
Sunae-Dong, 6-1, Bundang-Gu
SeongNam-City, Kyunggi-Do
Republic of Korea 463-825
Phone: +031.717.2780
Fax: +031.717.2782

China - Shanghai
9th Floor, Park Center
1088 Fangdian Road, Pudong New District
Shanghai 201204, China
Phone: +86.21.6088.0608
Fax: +86.21.6088.0366

Taiwan
Suite C1, No. 1, Li-Hsin Road 1
Science-Based Industrial Park
Hsinchu 30078, Taiwan
Phone: +886.3.666.8828
Fax: +886.3.666.1282

Japan - Yokohama
Shin-Yokohama Business Center Bldg. 5th Floor
3-2-6 Shin-Yokohama, Kohoku-ku,
Yokohama, Kanagawa, 222-0033, Japan
Phone: +81.45.548.6150
Fax: +81.45.548.6151

China - Shenzhen
Unit E, 5th Floor
No. 2 Finance Base
8 Ke Fa Road
Shenzhen, 518057, China
Phone: +86.755.3301.0366
Fax: +86.755.3301.0966

I Contents

II Preface

This document provides technical details using a set of consistent typographical conventions to help the user differentiate key concepts at a glance.

Conventions include:

Example	Description
AmbaGuiGen, **DirectUSB** **Save**, **File > Save** **Power**, **Reset**, **Home**	Software names GUI commands and command sequences Computer / Hardware buttons
Flash_IO_control **da**, **status**, **enable**	Register names and register fields. For example, **Flash_IO_control** is the register for global control of Flash I/O, and bit 17 (**da)** is used for DMA acknowledgement.
GPIO81, **CLK_AU**	Hardware external pins
VIL, VIH, VOL, VOH	Hardware pin parameters
INT_O, RXDATA_I	Hardware pin signals
amb_performance_t ***amb_operating_mode_t*** **amb_set_operating_mode()**	API details (e.g., functions, structures, and type definitions)

`/usr/local/bin`
`success = amb_set_operating_mode (amb_hal_base_address, & operating_mode)`
	User entries into software dialogues and GUI windows
	File names and paths
	Command line scripting and Code

Table II-1. Typographical Conventions for Technical Documents.

Additional Ambarella typographical conventions include:

- Acronyms are given in UPPER CASE using the default font (e.g., AHB, ARM11 and DDRIO).
- Names of Ambarella documents and publicly available standards, specifications, and databooks appear in *italic* type.

1 Overview

1.1 Overview: Introduction

Exchangeable image file format (Exif) is a standard that specifies the formats for images, sound, and ancillary tags used by digital cameras (including smartphones), scanners and other systems handling image and sound files recorded by digital cameras.
This document describes the Exif APIs supported by Ambarella image algorithm middleware (Image Algo).

1.2 Exif: Overview

The Exif-related APIs provide tools to set camera parameters; and based on these parameters, the user can compute the Exif information.

1.3 Exif: List of APIs

There are two main types of Exif APIs.

1. Camera Parameters APIs

   ```
   int AmbaImg_Exif_Set_GFNumberNum(UINT16 fNumberNum);
   int AmbaImg_Exif_Get_GFNumberNum(UINT16 *pFNumberNum);
   int AmbaImg_Exif_Set_GFocusDistanceNum(UINT32 focusDistanceNum);
   int AmbaImg_Exif_Get_GFocusDistanceNum(UINT32 *pFocusDistanceNum);
   int AmbaImg_Exif_Set_GFocalLength35mmFilm(UINT16 focalLength35mmFilm);
   int AmbaImg_Exif_Get_GFocalLength35mmFilm(UINT16 *pFocalLength35mmFilm);
   int AmbaImg_Exif_Set_GLensinfo(Exif_LENS_INFO_s *pSetGLensinfo);
   int AmbaImg_Exif_Get_GLensinfo(Exif_LENS_INFO_s *pGetGLensinfo);
   ```

2. Exif Information APIs

   ```
   void Amba_Img_Exif_Compute_AAA_Exif(COMPUTE_EXIF_PARAMS_s *pComputeExifParams);
   int Amba_Img_Exif_Get_Exif_Info(UINT8 aeIdx, EXIF_INFO_s *pExifInfo);
   ```

2 List of Related APIs

2.1 List of Related APIs: API Syntax

The chapter provides details on all APIs related to camera parameters and Exif information.

2.2 List of Related APIs: List of APIs

Camera Parameters:

- AmbaImg_Exif_Set_GFNumberNum
- AmbaImg_Exif_Get_GFNumberNum
- AmbaImg_Exif_Set_GFocusDistanceNum
- AmbaImg_Exif_Get_GFocusDistanceNum
- AmbaImg_Exif_Set_GFocalLength35mmFilm
- AmbaImg_Exif_Get_GFocalLength35mmFilm
- AmbaImg_Exif_Set_GLensinfo
- AmbaImg_Exif_Get_GLensinfo

Exif Information API:

- Amba_Img_Exif_Compute_AAA_Exif
- Amba_Img_Exif_Get_Exif_Info

2.3 List of Related APIs: Camera Parameters APIs

This section introduces the APIs of camera parameters.

2.3.1 AmbaImg_Exif_Set_GFNumberNum

API Syntax:

int **AmbaImg_Exif_Set_GFNumberNum** (UINT16 fNumberNum)

Function Description:

- This API is used to set the F-Number numerator value, and the denominator of F-Number is 100.
- It means that the user wants to set F-Number as 2.8; and then, set fNumberNum as 280.

Parameters:

Type	Parameter	Description
UINT16	**fNumberNum**	Set F-Number numerator

*Table 2-1. Parameters for Exif API **AmbaImg_Exif_Set_GFNumberNum()**.*

Returns:

Return	Description
0	Success
- 1	Failure

*Table 2-2. Returns for Exif API **AmbaImg_Exif_Set_GFNumberNum()**.*

Example:

None

See Also:

AmbaImg_Exif_Get_GFNumberNum()

2.3.2 AmbaImg_Exif_Get_GFNumberNum

API Syntax:

int **AmbaImg_Exif_Get_GFNumberNum** (UINT16 *pFNumberNum)

Function Description:

- This API is used to get the F-Number numerator value, and the denominator of F-Number is 100.
- Ex. If the user gets *pFNumberNum = 280, the real F Number is 280 divided by 100. It means the F-Number is 2.8.

Parameters:

Type	Parameter	Description
UINT16*	**pFNumberNum**	Get F-Number numerator value to *pFNumberNum.

Table 2-3. Parameters for Exif API ***AmbaImg_Exif_Get_GFNumberNum()****.*

Returns:

Return	Description
0	Success
- 1	Failure

Table 2-4. Returns for Exif API ***AmbaImg_Exif_Get_GFNumberNum()****.*

Example:

None

See Also:

AmbaImg_Exif_Set_GFNumberNum()

2.3.3 AmbaImg_Exif_Set_GFocusDistanceNum

API Syntax:

int **AmbaImg_Exif_Set_GFocusDistanceNum** (UINT32 focusDistanceNum)

Function Description:

- This API is used to set the focus distance numerator value, and the denominator of the focus distance is 100. It means that the real focus distance is focusDistanceNum divided by 100.

Parameters:

Type	Parameter	Description
UINT32*	**focusDistanceNum**	Set Focus distance numerator.

*Table 2-5. Parameters for Exif API **AmbaImg_Exif_Set_GFocusDistanceNum()**.*

Returns:

Return	Description
0	Success
- 1	Failure

*Table 2-6. Returns for Exif API **AmbaImg_Exif_Set_GFocusDistanceNum()**.*

Example:

None

See Also:

AmbaImg_Exif_Get_GFocusDistanceNum()

2.3.4 AmbaImg_Exif_Get_GFocusDistanceNum

API Syntax:

int **AmbaImg_Exif_Get_GFocusDistanceNum** (UINT32 *pFocusDistanceNum)

Function Description:

- This API is used to get the focus distance numerator, and the denominator of the focus distance is 100. It means the real focus distance is *pFocusDistanceNum divided by 100.

Parameters:

Type	Parameter	Description
UINT32*	**pFocusDistanceNum**	Get focus distance numerator value to *pFocusDistanceNum.

*Table 2-7. Parameters for Exif API **AmbaImg_Exif_Get_GFocusDistanceNum()**.*

Returns:

Return	Description
0	Success
- 1	Failure

*Table 2-8. Returns for Exif API **AmbaImg_Exif_Get_GFocusDistanceNum()**.*

Example:

None

See Also:

AmbaImg_Exif_Set_GFocusDistanceNum()

2.3.5 AmbaImg_Exif_Set_GFocalLength35mmFilm

API Syntax:

int **AmbaImg_Exif_Set_GFocalLength35mmFilm** (UINT16 focalLength35mmFilm)

Function Description:

- This API is used to set 35mm Film focal length.

Parameters:

Type	Parameter	Description
UINT16	**focalLength35mmFilm**	35mm Film focal length

*Table 2-9. Parameters for Exif API **AmbaImg_Exif_Set_GFocalLength35mmFilm()**.*

Returns:

Return	Description
0	Success
- 1	Failure

*Table 2-10. Returns for Exif API **AmbaImg_Exif_Set_GFocalLength35mmFilm()**.*

Example:

None

See Also:

AmbaImg_Exif_Get_GFocalLength35mmFilm()

2.3.6 AmbaImg_Exif_Get_GFocalLength35mmFilm

API Syntax:

int **AmbaImg_Exif_Get_GFocalLength35mmFilm** (UINT16 *pFocalLength35mmFilm)

Function Description:

- This API is used to get 35mm Film focal length.

Parameters:

Type	Parameter	Description
UINT16*	**pFocalLength35mmFilm**	Get 35mm Film focal length to *pFocalLength35mmFilm.

Table 2-11. Parameters for Exif API ***AmbaImg_Exif_Get_GFocalLength35mmFilm()***.

Returns:

Return	Description
0	Success
- 1	Failure

Table 2-12. Returns for Exif API ***AmbaImg_Exif_Get_GFocalLength35mmFilm()***.

Example:

None

See Also:

AmbaImg_Exif_Set_GFocalLength35mmFilm()

2.3.7 AmbaImg_Exif_Set_GLensinfo

API Syntax:

int **AmbaImg_Exif_Set_GLensinfo** (Exif_LENS_INFO_s *pSetGLensinfo)

Function Description:

- This API is used to set the lens information.

Parameters:

Type	Parameter	Description
Exif_LENS_INFO_s *	**pSetGLensinfo**	Set the lens information. Please refer to Section 2.3.7.1 for more details.

*Table 2-13. Parameters for Exif API **AmbaImg_Exif_Set_GLensinfo()**.*

Returns:

Return	Description
0	Success
- 1	Failure

*Table 2-14. Returns for Exif API **AmbaImg_Exif_Set_GLensinfo()**.*

Example

None

See Also:

AmbaImg_Exif_Get_GLensinfo()

2.3.7.1 AmbaImg_Exif_Set_GLensinfo > Exif_LENS_INFO_s

Type	Field	Description
UINT8	**FlashType**	Type of flash supported, ‘0’ if no flash supported.
UINT16	**AptValueMax**	100* valid aperture value

*Table 2-15. Definition of **Exif_LENS_INFO_s** for Exif API **AmbaImg_Exif_Set_GLensinfo()**.*

2.3.8 AmbaImg_Exif_Get_GLensinfo

API Syntax:

int **AmbaImg_Exif_Get_GLensinfo (**Exif_LENS_INFO_s *pGetGLensinfo**)**

Function Description:

- This API is used to get the lens information.

Parameters:

Type	Parameter	Description
Exif_LENS_ INFO_s *	**pSetGLensinfo**	Lens information. Please refer to Section 2.3.7.1 for more details.

*Table 2-16. Parameters for Exif API **AmbaImg_Exif_Get_GLensinfo()**.*

Returns:

Return	Description
0	Success
- 1	Failure

*Table 2-17. Returns for Exif API **AmbaImg_Exif_Get_GLensinfo()**.*

Example:

None

See Also:

AmbaImg_Exif_Set_GLensinfo()

2.4 List of Related APIs: Exif Information API Syntax

This section introduces APIs related to Exif Information.

2.4.1 Amba_Img_Exif_Compute_AAA_Exif

API Syntax:

void **Amba_Img_Exif_Compute_AAA_Exif** (COMPUTE_EXIF_PARAMS_s *pComputeExifParams)

Function Description:

- This API is used to compute Exif information.

Parameters:

Type	Parameter	Description
COMPUTE_EXIF_PARAMS_s *	**pComputeExifParams**	Set to compute Exif parameters. Please refer to Section 2.4.1.1 for more details.

*Table 2-18. Parameters for Exif API **Amba_Img_Exif_Compute_AAA_Exif()**.*

Returns:

None

Example:

None

See Also:

None

2.4.1.1 Amba_Img_Exif_Compute_AAA_Exif > COMPUTE_EXIF_PARAMS_s

Type	Field	Description
UINT32	**ChannelNo**	Channel number
UINT32	**Mode**	Define the capture mode: **IMG_EXIF_STILL**, **IMG_EXIF_PIV**, **IMG_EXIF_PB**, **IMG_EXIF_LASTMODE**.
UINT8	**AeIdx**	AE index

*Table 2-19. Definition of **COMPUTE_EXIF_PARAMS_s** for Exif API **Amba_Img_Exif_Compute_AAA_Exif()**.*

2.4.2 Amba_Img_Exif_Get_Exif_Info

API Syntax:

int **Amba_Img_Exif_Get_Exif_Info** (UINT8 aeIdx, EXIF_INFO_s *pExifInfo)

Function Description:

- This API is used to get the Exif information.
- There are many parameters expressed as numerators and denominators.
- For example: ExposureTimeNum as numerator, ExposureTimeDen as denominator. When ExposureTimeNum is divided by ExposureTimeDen, the user can get the real Exposure Time. The other numerators and denominators have a similar relationship.

Parameters:

Type	Parameter	Description
UINT8	**aeIdx**	AE index
EXIF_INFO_s *	**pExifInfo**	Get Exif information to *pExifInfo. Please refer to Section 2.4.2.1 for more details.

*Table 2-20. Parameters for API **Amba_Img_Exif_Get_Exif_Info()**.*

Returns:

Return	Description
0	Success
- 1	Failure

*Table 2-21. Returns for API **Amba_Img_Exif_Get_Exif_Info()**.*

Example:

Before getting Exif information, the user should compute the Exif information. It means that the user should call " Amba_Img_Exif_Compute_AAA_Exif(COMPUTE_EXIF_PARAMS_s *pComputeExifParams)" at first.

Step1. Assign Exif parameters
pComputeExifParams-> ChannelNo = XXXX
pComputeExifParams-> Mode = XXXX
pComputeExifParams-> AeIdx = XXXX

Step2. Compute Exif parameters
Amba_Img_Exif_Compute_AAA_Exif(COMPUTE_EXIF_PARAMS_s *pComputeExifParams);

Step3. Assign Exif parameters
get which one shot by selecting aeIdx.

```
  Amba_Img_Exif_Get_Exif_Info(UINT8 aeIdx, EXIF_INFO_s *pExifInfo);
  after calling this function, the exif information will be saved in *pExifInfo
```

See Also:

Amba_Img_Exif_Compute_AAA_Exif()

2.4.2.1 Amba_Img_Exif_Get_Exif_Info > EXIF_INFO_s

Type	Field	Description
UINT32	**ExposureTimeNum**	Numerator of the exposure time
UINT32	**ExposureTimeDen**	Denominator of the exposure time, default value is 1000000.
UINT32	**FNumberNum**	Numerator of f number
UINT32	**FNumberDen**	Denominator of f number, default value is 100.
UINT32	**ExposureProgram**	The class of the program used by the camera to set the exposure when the picture is taken.
UINT32	**FocalLengthNum**	Numerator of focal length
UINT32	**FocalLengthDen**	Denominator of focal length, default value is 100.
UINT32	**SubjectDistanceNum**	Numerator of the subject distance
UINT32	**SubjectDistanceDen**	Denominator of the subject distance, default value is 100.
UINT16	**CustomRendered**	This tag indicates the use of special processing on image data such as rendering geared to output.
UINT16	**ExposureMode**	This tag indicates the exposure mode set when the image was shot.
UINT16	**WhiteBalance**	This tag indicates the white balance mode set when the image was shot.
UINT16	**FocalLength35mmFilm**	This tag indicates the equivalent focal length assuming a 35mm film camera, in mm.
UINT16	**SceneCaptureType**	This tag indicates the type of scene that was shot.
UINT16	**Contrast**	This tag indicates the direction of contrast processing applied by the camera.
UINT16	**Saturation**	This tag indicates the direction of saturation processing applied by the camera.
UINT16	**Sharpness**	This tag indicates the direction of sharpness processing applied by the camera.
UINT16	**SubjectDistanceRange**	This tag indicates the distance to the subject.
INT32	**ShutterSpeedNum**	Numerator of shutter speed
INT32	**ShutterSpeedDen**	Denominator of shutter speed, default value is -1000.
UINT32	**ApertureValueNum**	Numerator of aperture value
UINT32	**ApertureValueDen**	Denominator of aperture value, default value is 100.
INT16	**ExposureBiasValueNum**	Numerator of exposure bias value
INT16	**ExposureBiasValueDen**	Denominator of exposure bias value, default value is 32.
UINT32	**MaxApertureValueNum**	Numerator of maximum aperture
UINT32	**MaxApertureValueDen**	Denominator of maximum aperture, default value is 100.
UINT32	**DigitalZoomRatioNum**	Numerator of digital zoom ratio
UINT32	**DigitalZoomRatioDen**	Denominator of digital zoom ratio, default value is 1024.
UINT16	**MeteringMode**	The metering mode
UINT16	**LightSource**	The kind of light source
UINT8	**FlashPixVersion[4]**	The flash pix format version
UINT16	**ColorSpace**	Color space specifier
UINT16	**SensingMethod**	Indicates the image sensor type on the camera or input device.

Type	Field	Description
UINT32	**ExposureIndexNum**	Numerator of exposure index
UINT32	**ExposureIndexDen**	Denominator of exposure index
UINT16	**FileSource**	Indicates the image source.
UINT16	**SceneType**	Indicates the type of scene.
UINT16	**Flash**	Flash
UINT16	**IsoSpeedRating**	IsoSpeedRating
UINT8	**GainControl**	This tag indicates the degree of the overall image gain adjustment.

*Table 2-22. Definition of **EXIF_INFO_s** for Exif API **Amba_Img_Exif_Get_Exif_Info()**.*

Appendix 1 Additional Resources

Please contact an Ambarella representative for digital copies.

Appendix 2 Important Notice

All Ambarella design specifications, datasheets, drawings, files, and other documents (together and separately, “materials”) are provided on an “*as is*” basis, and Ambarella makes no warranties, expressed, implied, statutory, or otherwise with respect to the materials, and expressly disclaims all implied warranties of noninfringement, merchantability, and fitness for a particular purpose. The information contained herein is believed to be accurate and reliable. However, Ambarella assumes no responsibility for the consequences of use of such information.

Ambarella Incorporated reserves the right to correct, modify, enhance, improve, and otherwise change its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

All products are sold subject to Ambarella’s terms and conditions of sale supplied at the time of order acknowledgment. Ambarella warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with its standard warranty. Testing and other quality control techniques are used to the extent Ambarella deems necessary to support this warranty.

Ambarella assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using Ambarella components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

Ambarella does not warrant or represent that any license, either expressed or implied, is granted under any Ambarella patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which Ambarella products or services are used. Information published by Ambarella regarding third-party products or services does not constitute a license from Ambarella to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from Ambarella under the patents or other intellectual property of Ambarella.

Reproduction of information from Ambarella documents is not permissible without prior approval from Ambarella.

Ambarella products are not authorized for use in safety-critical applications (such as life support) where a failure of the product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Customers acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of Ambarella products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by Ambarella. Further, Customers must fully indemnify Ambarella and its representatives against any damages arising out of the use of Ambarella products in such safety-critical applications.

Ambarella products are neither designed nor intended for use in automotive and military/aerospace applications or environments. Customers acknowledge and agree that any such use of Ambarella products is solely at the Customer’s risk, and they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

Appendix 3 Revision History

NOTE: Page numbers for previous drafts may differ from page numbers in the current version.

Version	Date	Comments
0.1	3 June 2014	Formatting into InDesign.
0.2	23 December 2014	Formatted into SDK6.

Table A3-1. Revision History.