

**Face API** ..... 1

**FaceEx API** ..... 1

..... 1

BS2Face ..... 1

BS2AuthGroup ..... 2

BS2TemplateEx ..... 2

BS2FaceEx ..... 3

# Face API

API

- [BS2\\_ScanFace](#): IR Face
- [BS2\\_GetAuthGroup](#): 가
- [BS2\\_GetAllAuthGroup](#): 가
- [BS2\\_SetAuthGroup](#):
- [BS2\\_RemoveAuthGroup](#):
- [BS2\\_RemoveAllAuthGroup](#):

## FaceEx API

- [BS2\\_ScanFaceEx](#): Visual Face  
[+ 2.7.1]
- [BS2\\_ExtractTemplateFaceEx](#): Visual Face 가 template  
[+ 2.7.1]
- [BS2\\_GetNormalizedImageFaceEx](#): Visual Face WARP (가  
) , WARP [+ 2.8]

## BS2Face

```
typedef struct {
    uint8_t faceIndex;
    uint8_t numOfTemplate;
    uint8_t flag;
    uint8_t reserved;

    uint16_t imageLen;
    uint8_t reserved2[2];

    uint8_t imageData[BS2_FACE_IMAGE_SIZE];
    uint8_t templateData[BS2_TEMPLATE_PER_FACE][BS2_FACE_TEMPLATE_LENGTH];
} BS2Face;
```

1. *faceIndex*

2. *numOfTemplate*

3. *flag*

FaceStation2 V1.4.0, FaceLite V1.2.0

flag

0

4. *reserved*

5. *imageLen*

6. *reserved2*

4. *imageData*

5. *data*

## BS2AuthGroup

```
typedef struct {
    BS2_AUTH_GROUP_ID    id;
    char                 name[BS2_MAX_AUTH_GROUP_NAME_LEN];
    uint8_t              reserved[32];
} BS2AuthGroup;
```

1. *id*

2. *name*  
BioStar

3. *reserved*

## BS2TemplateEx

```
typedef struct {
    uint8_t              data[552];
    uint8_t              isIR;
    uint8_t              reserved[3];
} BS2TemplateEx;
```

Visual Face

1. *data*  
IR visual

2. *isIR*  
IR true, visual false

### 3. reserved

## BS2FaceEx

```

typedef struct {
    uint8_t      faceIndex;
    uint8_t      numOfTemplate;
    uint8_t      flag;
    uint8_t      reserved;

    uint32_t     imageLen;
    union {
        struct {
            uint16_t irImageLen;
            uint8_t  unused[6];           ///< 6 bytes (packing)
            uint8_t  imageData[BS2_MAX_WARPED_IMAGE_LENGTH];  ///<
40 * 1024 bytes
            uint8_t  irImageData[BS2_MAX_WARPED_IR_IMAGE_LENGTH];  ///<
30 * 1024 bytes
            BS2TemplateEx templateEx[BS2_MAX_TEMPLATES_PER_FACE_EX];  ///<
20 * 556 bytes
        };

        uint8_t     *rawImageData;

        BS2TemplateEx *onlyTemplateEx;
    };
} BS2FaceEx;

```

### Visual Face

1. faceIndex

2. numOfTemplate

Visual, IR

3. flag

가 WARP  
WARP , , ,

flag가 1 , union struct 5

flag가 0 , union rawImageData  
가 WARP 가 , ,

flag BS2\_FACE\_EX\_FLAG\_NONE(0) , rawImageData  
imageLen , ,

rawImageData 가 WARP struct

rawImageData struct가 union

[+ 2.9.6]

BS2\_FACE\_EX\_FLAG\_TEMPLATE\_ONLY(0x20)

가

가

numOfTemplate

onlyTemplateEx

가

, BS2FaceConfig unableToSaveImageOfVisualFace 1(true)

BS2_FACE_EX_FLAG_NONE	0x00
BS2_FACE_EX_FLAG_WARPED	0x01
BS2_FACE_EX_FLAG_TEMPLATE_ONLY	0x20
BS2_FACE_EX_FLAG_ALL	0xFF

4. reserved

5. imageLen

6. irlImageLen

IR

7. unused

(packing )

8. imageData

WARP

. rawData

		FW			rawImageData
FaceStation F2	V1	2.0.0	250 * 250	JPG	JPG
FaceStation F2	V2	2.0.0	112 * 112	PNG	JPG, PNG
BioStation 3	V1		112 * 112	PNG	JPG, PNG

9. irlImageData

IR

. WARP

IR

10. templateEx

Visual

IR

11. rawData

WARP

. imageData

WARP

가

4000

1920

12. onlyTemplateEx

[+ 2.9.6]

BS2TemplateEx

, numOfTemplate

flag가 BS2\_FACE\_EX\_FLAG\_TEMPLATE\_ONLY

From:

<https://kb.supremainc.com/kbtest/> - **BioStar 2 Device SDK**

Permanent link:

[https://kb.supremainc.com/kbtest/doku.php?id=ko:face\\_api](https://kb.supremainc.com/kbtest/doku.php?id=ko:face_api)

Last update: **2024/10/24 14:31**