

**BS2\_ExtractTemplateFaceEx** ..... 1  
..... 1  
..... 1  
..... 1  
..... 1

# BS2\_ExtractTemplateFaceEx

[+ 2.7.1] FaceStation F2 AOC template 가 , template .

```
#include "BS_API.h"

int BS2_ExtractTemplateFaceEx(void* context, uint32_t deviceId, const
uint8_t* imageData, uint32_t imageDataLen, bool isWarped, BS2TemplateEx*
templateEx);
```

## BS2TemplateEx

- [In] *context* : Context
- [In] *deviceId* :
- [In] *imageData* : 가
- [In] *imageDataLen* : imageData
- [In] *isWarped* : warp
- [Out] *templateEx* : imageData가 가 template

BS\_SDK\_SUCCESS , 가

C++

```
int UserControl::extractTemplateFaceEx(BS2_DEVICE_ID id, BS2TemplateEx&
templateEx)
{
    BS2SimpleDeviceInfoEx deviceInfoEx = { , };

    int sdkResult = BS2_GetDeviceInfoEx(context_, id, NULL, &deviceInfoEx);
```

```
if (BS_SDK_SUCCESS != sdkResult)
{
    TRACE("BS2_GetDeviceInfoEx call failed: %d", sdkResult);
    return sdkResult;
}

bool faceExScanSupported = (deviceInfoEx.supported &
BS2SimpleDeviceInfoEx::BS2_SUPPORT_FACE_EX_SCAN) ==
    BS2SimpleDeviceInfoEx::BS2_SUPPORT_FACE_EX_SCAN;
if (faceExScanSupported)
{
    try
    {
        if (Utility::isYes("Do you want to extract faceEx template from
image?"))
        {
            string imagePath = Utility::getInput<string>("Enter the face
image path and name:");//C:\88withphone.jpg
            uint32_t size = Utility::getResourceSize(imagePath);
            shared_ptr<uint8_t> buffer(new uint8_t[size],
ArrayDeleter<uint8_t>());

            size_t dataOffset = offsetof(BS2FaceEx, rawImageData);
            size_t faceSize = dataOffset + size;
            if (Utility::getResourceFromFile(imagePath, buffer, size))
            {
                sdkResult = BS2_ExtractTemplateFaceEx(context_, id,
buffer.get(), size, , &templateEx);
                if (BS_SDK_SUCCESS != sdkResult)
                {
                    TRACE("BS2_ExtractTemplateFaceEx call failed: %d",
sdkResult);

                    return sdkResult;
                }
                print(templateEx);
            }
        }
    }
    catch (const std::exception&)
    {
    }
}

return sdkResult;
}
BS2_ReleaseObject(uidObj);
```

From:

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

Permanent link:

[https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2\\_extracetablefaceex&rev=1656541279](https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2_extracetablefaceex&rev=1656541279)

Last update: **2022/06/30 07:21**