



# NBioAPI Image Converter Specification

Ver 1.00

Software Development Department

NITGEN Co., Ltd.

Document Version History

Version	Date	Comments
1.00	12-MAR-2004	Initial Release

# 1. Overview

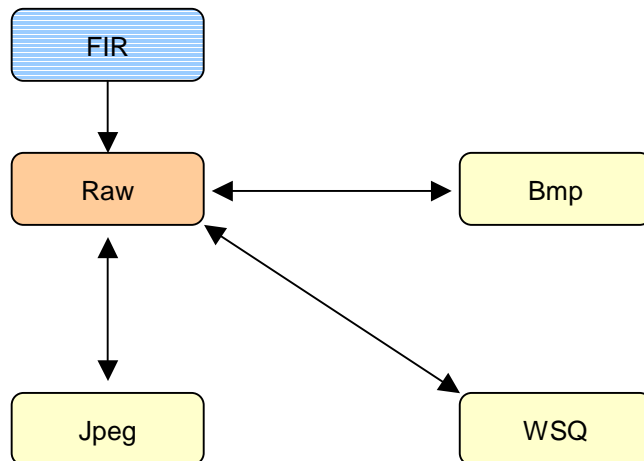
This document describes some APIs that can be used to convert “raw” images, captured by NITGEN HFUDU01/MFUDU01 devices, into other format of image data such as bmp, jpeg and WSQ.

File types supported are as below.

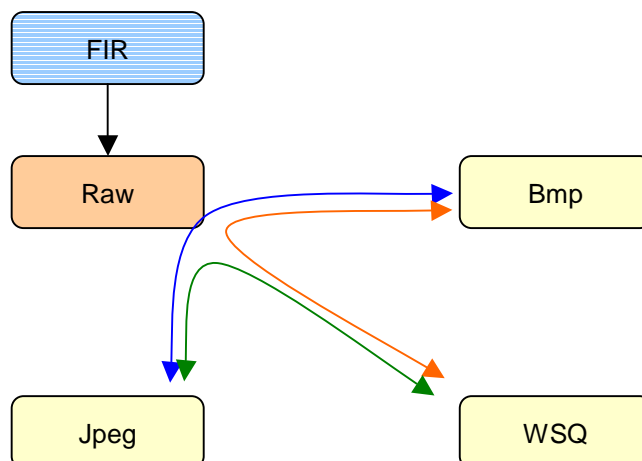
- Raw Buffer → Bmp Buffer
- Bmp Buffer → Raw Buffer
- Raw Buffer → Jpeg Buffer
- Jpeg Buffer → Raw Buffer
- Raw Buffer → WSQ Buffer
- WSQ Buffer → Raw Buffer

## 2. Diagram

Image conversion must be made out of a "raw" image retrieved from FIR data, supporting buffer-to-buffer transaction only at this version. Following diagram indicates all possible cases of data interaction between different image data types.



Use of some API combination is necessary to complete a data conversion between two non-raw image data. For example, a bmp image data can be converted to one of jpeg format by using the both NBioAPI\_ImgConvBmpToRawBuf and the NBioAPI\_ImgConvRawToJpgBuf functions.



## 3. Functions

### ■ NBioAPI\_ImgConvRawToBmpBuf

```
NBioAPI_RETURN NBioAPI NBioAPI_ImgConvRawToBmpBuf(  
    IN LPBYTE    lpImageBuffer,  
    IN  UINT      nWidth,  
    IN  UINT      nHeight,  
    OUT LPBYTE    lpBMPBuffer,  
    OUT int *     nBMPBufLen);
```

#### Description

This function is to convert a “raw” image data to one of bmp format.

#### Parameters

lpImageBuffer:

A pointer to a BYTE containing a “raw” image data.

nWidth :

An integer value indicating the “raw” image width in pixel. (248 for the HFDU01 device)

nHeight :

An integer value indicating the “raw” image height in pixel. (292 for the HFDU01 device)

lpBMPBuffer :

A pointer to a BYTE that receives a bmp image data.

nBMPBufLen :

A pointer to an integer that receives the size of the bmp image data.

#### Return Value

NBioAPIERROR_NONE	: No error.
NBioAPIERROR_IMGCONV_INVALID_PARAM	: Invalid parameter.
NBioAPIERROR_IMGCONV_MEMALLOC_FAIL	: Memory allocation failed.

**■ NBioAPI\_ImgConvBmpToRawBuf**

```
NBioAPI_RETURN NBioAPI NBioAPI_ImgConvBmpToRawBuf(  
    IN  LPBYTE    lpImageBuffer,  
    OUT UINT *    nWidth,  
    OUT UINT *    nHeight,  
    OUT LPBYTE    lpRawBuffer);
```

**Description**

This function is to convert a bmp image data to one of “raw” format.

**Parameters**

lpImageBuffer :

A pointer to a BYTE containing a bmp image data.

nWidth :

A pointer to an UINT that receives the “raw” image width, in pixel, that converted from a bmp image data. Using the HFDU01 device returns 248.

nHeight :

A pointer to an UINT that receives the “raw” image height, in pixel, that converted from a bmp image data. Using the HFDU01 device returns 292.

lpRawBuffer :

A pointer to a BYTE that receives a “raw” image data.

**Return Value**

NBioAPIERROR_NONE	: No error.
NBioAPIERROR_IMGCONV_INVALID_PARAM	: Invalid parameter.
NBioAPIERROR_IMGCONV_MEMALLOC_FAIL	: Memory allocation failed.

## ■ NBioAPI\_ImgConvRawToJpgBuf

```
NBioAPI_RETURN NBioAPI NBioAPI_ImgConvRawToJpgBuf(  
    IN  LPBYTE    lpRawBuffer,  
    IN  UINT       nWidth,  
    IN  UINT       nHeight,  
    IN  int        nQuality,  
    OUT LPBYTE    lpJpgBuffer,  
    OUT int *      nJpgBufLen);
```

### Description

This function is to convert a “raw” image data to one of jpeg format.

### Parameters

lpRawBuffer :

A pointer to a BYTE containing a “raw” image data.

nWidth :

An integer value indicating the “raw” image width in pixel. (248 for the HFDU01 device)

nHeight :

An integer value indicating the “raw” image height in pixel. (292 for the HFDU01 device)

nQuality :

An integer value indicating the compression rate to create a jpeg data. Using a value of 100 creates a jpeg data larger than “raw” data.

lpJpgBuffer :

A pointer to a BYTE that receives a jpeg image data.

nJpgBufLen :

A pointer to an integer that receives the size of the jpeg image data.

### Return Value

NBioAPIERROR_NONE	: No error.
NBioAPIERROR_IMGCONV_INVALID_PARAM	: Invalid parameter.
NBioAPIERROR_IMGCONV_MEMALLOC_FAIL	: Memory allocation failed.

## ■ NBioAPI\_ImgConvJpgToRawBuf

```
NBioAPI_RETURN NBioAPI NBioAPI_ImgConvJpgToRawBuf(  
    IN  LPBYTE    lpJpgBuffer,  
    IN  UINT       nJpgBufLen,  
    OUT UINT *     nWidth,  
    OUT UINT *     nHeight,  
    OUT LPBYTE    lpRawBuffer);
```

### Description

This function is to convert a jpeg image data to one of “raw” format.

### Parameters

lpJpgBuffer :

A pointer to a BYTE containing a jpeg image data.

nJpgBufLen :

An integer value indicating the size of jpeg buffer. This value must be identical to the JpgBufLen value returned from the NBioAPI\_ImgConvRawToJpgBuf function.

nWidth :

A pointer to an UINT that receives the “raw” image width, in pixel, that converted from a jpeg image data. Using the HFDU01 device returns 248.

nHeight :

A pointer to an UINT that receives the “raw” image height, in pixel, that converted from a jpeg image data. Using the HFDU01 device returns 292.

lpRawBuffer :

A pointer to a BYTE that receives a “raw” image data.

### Return Value

NBioAPIERROR_NONE	: No error.
NBioAPIERROR_IMGCONV_INVALID_PARAM	: Invalid parameter.
NBioAPIERROR_IMGCONV_MEMALLOC_FAIL	: Memory allocation failed.

## ■ NBioAPI\_ImgConvRawToWSQBuf

```
NBioAPI_RETURN NBioAPI NBioAPI_ImgConvRawToWSQBuf(  
    IN LPBYTE    lpRawBuffer,  
    IN int        nWidth,  
    IN int        nHeight,  
    OUT LPBYTE    lpWSQBuffer,  
    OUT int *     nReturn_size,  
    IN float      q);
```

### Description

This function is to convert a “raw” image data to one of WSQ format.

### Parameters

lpRawBuffer :

A pointer to a BYTE containing a “raw” image data.

nWidth :

An integer value indicating the “raw” image width in pixel. (248 for the HFDU01 device)

nHeight :

An integer value indicating the “raw” image height in pixel. (292 for the HFDU01 device)

lpWSQBuffer :

A pointer to a BYTE that receives a WSQ image data.

nReturn\_size :

A pointer to an integer that receives the size of the WSQ image data.

q :

A value indicating the quality of the WSQ image data, ranging from 1.0 to 98.0. The default value is 15.0, meaning 15:1. A value of 50.0 or more may cause the image to be distorted.

### Return Value

NBioAPIERROR_NONE	: No error.
NBioAPIERROR_IMGCONV_INVALID_PARAM	: Invalid parameter.
NBioAPIERROR_IMGCONV_MEMALLOC_FAIL	: Memory allocation failed.

## ■ NBioAPI\_I mgConvWSQToRawBuf

```
NBioAPI_RETURN NBioAPI NBioAPI_I mgConvWSQToRawBuf(  
    IN LPBYTE    lpWSQBuffer,  
    OUT LPBYTE    lpRawBuffer,  
    OUT int *     nReturn_size,  
    OUT int *     nWidth,  
    OUT int *     nHeight);
```

### Description

This function is to convert a WSQ image data to one of “raw” format.

### Parameters

lpWSQBuffer :

A pointer to a BYTE containing a WSQ image data.

lpRawBuffer :

A pointer to an UINT that receives the “raw” image width, in pixel, that converted from a WSQ image data.

nReturn\_size :

A pointer to an UINT that receives the “raw” image height, in pixel, that converted from a WSQ image data.

nWidth :

A pointer to an UINT that receives the “raw” image width, in pixel, that converted from a WSQ image data. Using the HFDU01 device returns 248.

nHeight :

A pointer to an UINT that receives the “raw” image height, in pixel, that converted from a WSQ image data. Using the HFDU01 device returns 292.

### Return Value

NBioAPIERROR_NONE	: No error.
NBioAPIERROR_IMGCONV_INVALID_PARAM	: Invalid parameter.
NBioAPIERROR_IMGCONV_MEMALLOC_FAIL	: Memory allocation failed.

[End of Doc]