

# ImageDb Programming Manual

2015-03-20

Version 1.8

**The content of this document is highly confidential  
and should be handled accordingly.**

**Confidential**

**These coded instructions, statements, and computer programs contain proprietary information of Nintendo and/or its licensed developers and are protected by national and international copyright laws. They may not be disclosed to third parties or copied or duplicated in any form, in whole or in part, without the prior written consent of Nintendo.**

## Table of Contents

---

1	Introduction .....	4
2	Configuring Image Files to Add.....	5
2.1	Unique Title ID .....	5
2.2	Available Pixel Resolutions for Images to Add .....	5
2.3	Displaying Images in Nintendo 3DS Camera .....	5
2.4	Saving Screenshots.....	6
3	Using Loaded Images.....	7
3.1	Focus Adjustment Values and Display Positions .....	7
4	Error Handling.....	8
4.1	Errors That Require User Notification.....	8
4.2	Message Examples.....	8
5	Notes.....	9
5.1	Control Files.....	9
5.2	Mounting SD Cards.....	9
5.3	Error Handling During Mounting .....	9
5.4	Using Threads.....	10
5.5	Restrictions on Handled Images .....	10
5.6	Image File Paths .....	11
5.7	When Loading Images Other Than Photos Taken With the Nintendo 3DS Camera .....	12
	Revision History .....	13

## Code

---

Code 1-1	Code to Add to the RSF .....	4
Code 5-1	Error Check Code for Mounting Failure .....	9

## Tables

---

Table 2-1	Displaying Images in the Nintendo 3DS Camera .....	6
Table 4-1	Error Message Examples.....	8
Table 5-1	Restrictions on Handled Images.....	11

# 1 Introduction

This document describes how to use the `ImageDb` library.

The `ImageDb` library reads from and adds to the images managed by Nintendo 3DS Camera on an SD Card. It also handles SD Card mounting and file reading and writing, so basically the application only has to deal with the transfer of RGB or YUV data.

Applications that use photos must also follow the *UGC Guidelines*.

The following lines must be added to the RSF (ROM specification) file for applications that use this library.

## Code 1-1 Code to Add to the RSF

```
AccessControlInfo:  
  FileSystemAccess:  
    - DirectSdmc
```

## 2 Configuring Image Files to Add

### 2.1 Unique Title ID

---

You must configure the 20-bit unique ID assigned to each title before you can add an image using the `ImageDb` library. Call the `imgdb::JpegMpBaseSaver::SetTitleUniqueId` function to set this value.

This information is recorded in images and has a variety of uses, such as filtering the images displayed by the Photo Selector applet.

To handle a saved title as one title even if it has been assigned multiple unique IDs for different markets and versions, set one of those assigned IDs as a common value.

### 2.2 Available Pixel Resolutions for Images to Add

---

You can use the `ImageDb` library to add images with resolutions within the range of 160×120 to 4096×3072 pixels. The width and height must each be a multiple of 16 pixels.

When an image is loaded with `ImageDb`, the image is shrunk by a 1/2, 1/4, 1/8, or 1/16 pattern, depending on the size of the image, to conform to a size of 160×120 to 1024×769.

### 2.3 Displaying Images in Nintendo 3DS Camera

---

The **View Photos** screen in Nintendo 3DS Camera displays images unchanged if they use no more pixels than can fit within the screen resolution. Any image that is too big to fit the screen is displayed zoomed out so that the entire image fits on the screen.

However, you can also display an image 80 pixels wider than the screen in order to adjust the 3D focus, just as you can for photos captured with Nintendo 3DS Camera. In such cases, call the `SetAdjustablePictureFlag` function and specify `true` as the argument.

Table 2-1 Displaying Images in the Nintendo 3DS Camera

Image Size	Display Focus Adjustment	Focus Adjustment	Margins	Display Method
Width greater than 400 or height greater than 240	OFF	No	Yes	 <p>Shrunk so that the width does not exceed 400 and the height does not exceed 240</p>
	ON (see <b>Note 1</b> )	Yes (Only MP files) (see <b>Note 2</b> )	No	 <p>Shrunk so that the width is 480</p>
Smaller than 400x240	Disabled	No	Yes	 <p>Unchanged</p>

**Note 1:** This display method is only enabled when the width is 480 or greater, the height is 240 or greater, and the aspect ratio is at least 2:1.

**Note 2:** An MP file contains image data in MP (multi-picture) format. The `ImageDb` library uses MP files to store 3D pictures (the files have an `.mpo` extension). For details about this format, see the *CTR-SDK API Reference*.

## 2.4 Saving Screenshots

Call the `SetScreenshotFlag` function when saving a screenshot of the image displayed onscreen. This allows you to display it with the special "game theme" for slide shows in Nintendo 3DS Camera.

Applications that save three-dimensional screenshots must save only JPEG images when stereoscopic images are not displayed. Specifically, use `imgdb::MpSaver` when `nn::gx::IsStereoVisionAllowed` returns `true`, and `imgdb::JpegSaver` when it returns `false`.

This is intended to prevent Nintendo 3DS Camera from attempting to use 3D display for images that cannot be viewed stereoscopically. This therefore does not apply to 3D imagery such as camera images and pre-rendered movies, or generated 3D imagery, for which you cannot adjust the parallax.

## 3 Using Loaded Images

### 3.1 Focus Adjustment Values and Display Positions

---

With the Nintendo 3DS cameras, focus is adjusted by displaying the images from the left and right cameras as shifted horizontally, so that images won't appear to be jumping out at you.

When using ImageDb to display loaded images, normally display images with a shift to the left and right corresponding to the value of `GetOffset (H)`.

This value indicates an amount of movement in terms of pixels for the image, and needs to be enlarged or shrunk depending on the size at which the image is displayed. In addition, note that the parallax does not become too great when the display position of the image overall is different on the right and left screens (for layouts that place the 3D picture itself in the position that jumps out to you).

If there are features that automatically adjust to an appropriate position or a method for the user to make adjustments easily, as long as the parallax won't become too great, there is no need to use this value.

## 4 Error Handling

### 4.1 Errors That Require User Notification

Many of the `ImageDb` library functions return a value of type `imgdb::Result`.

The application must inform the user of the error details for the following result types.

- `ResultErrorNotExistStorage`
- `ResultErrorWriteProtect`
- `ResultErrorInsufficientStorageSpace`
- `ResultErrorOverNumImages`
- `ResultErrorNotAvailableStorage`

### 4.2 Message Examples

The following table gives some sample messages. Change these as appropriate for your application's use case.

**Table 4-1 Error Message Examples**

Result Value	Message Examples
<code>ResultErrorNotExistStorage</code>	SD Card is not inserted.
<code>ResultErrorWriteProtect</code>	The SD Card is write-protected.
<code>ResultErrorInsufficientStorageSpace</code>	Could not save as SD Card does not have sufficient free space available.
<code>ResultErrorOverNumImages</code>	Could not save image as SD Card already has maximum number of images.
<code>ResultErrorNotAvailableStorage</code>	Could not recognize SD Card.
Anything else:	SD Card operation failed.

**Note:** You may display the same message for return values of both `ResultErrorOverNumImages` and `ResultErrorInsufficientStorageSpace` when you use `imgdb::util::EstimateRemainPictureNum` to check for errors in advance.

**Note:** There is a file in `tools/ResultErrorOverNumImages_sample` in the `ImageDb` package in which dummy data has already been recorded up to the limit. Copy this to the `Nintendo 3DS/Private` folder on the SD card to use it.

## 5 Notes

### 5.1 Control Files

---

The `ImageDb` library and Nintendo 3DS Camera create control files on an SD Card for faster searching of managed images and movies (AVI files).

The `ImageDb` library updates the control files when images are added. The library even fixes control files if they are corrupted, so applications normally do not need to pay much attention to them.

The system can manage up to a total of 3000 image and movie files on any single SD Card. Attempting to add any files in excess of this maximum causes the called function to return a `ResultErrorOverNumImages` error. This is the same as the maximum number that can be displayed by Nintendo 3DS Camera and the Photo Selector applet.

### 5.2 Mounting SD Cards

---

The `ImageDb` library provides the `imgdb::Initialize` function to mount an SD Card, and the `imgdb::Finalize` function to unmount an SD Card. This is not affected by mount operations with the `nn::fs::MountSdmc` function and is managed separately.

Call the `imgdb::mnt::MountSdmc` and `imgdb::mnt::UnmountSdmc` functions to control the mount state at any time. However, be sure to call these functions as a pair.

You can call these functions to handle hot swapping when the library is notified by functions such as `nn::fs::RegisterSdmcInsertedEvent`.

### 5.3 Error Handling During Mounting

---

If an SD Card fails to mount when inserted, you can check the SD Card's state with `imgdb::mnt::CheckSdmcState`. However, you might not obtain the correct result if the SD Card is removed before this function call is completed.

To avoid this problem, include some type of handling such as ignoring the result if an SD Card removal is detected, as shown below.

#### Code 5-1 Error Check Code for Mounting Failure

```
// Initialize by automatically resetting
sdInsertedEvent.Initialize(false);
sdEjectedEvent.Initialize(false);

// State check
bool fInserted = sdInsertedEvent.TryWait();
bool fEjected = sdEjectedEvent.TryWait(); // Flag is cleared here
if(fInserted || fEjected)
```

```

{
    if(nn::fs::IsSdmcInserted())
    {
        bool fResult = imgdb::mnt::MountSdmc();
        if(fResult == false)
        {
            eResult = imgdb::mnt::CheckSdmcState();
            if(sdEjectedEvent.TryWait()) // Checks whether SD Card was removed
                                        // during this interval
            {
                // Fails because SD Card was removed
            }
            else
            {
                // Carry out ImageDb error process (eResult)
            }
        }
    }
    else
    {
        imgdb::mnt::UnmountSdmc();
    }
}

```

## 5.4 Using Threads

Because `ImageDb` is not thread-safe, there is a possibility of abnormal operations if various API functions are called from multiple threads.

A mutex is required for the allocator passed to the `imgdb::Initialize` function from other threads.

If only a single thread handles `ImageDb`, it is acceptable to simply prepare a dedicated heap for use by the allocator.

## 5.5 Restrictions on Handled Images

There are restrictions on the images that can be handled by `ImageDb` and the system, and these are described in the table below.

Also note that because `ImageDb` and the system use the CTR-SDK's `nn::jpeg` library, some images exist that meet the restrictions below but still cannot be handled. For details, see the *CTR-SDK API Reference*.

**Table 5-1 Restrictions on Handled Images**

	Added to ImageDb	Loaded to ImageDb	Photo Selector Applet	Nintendo 3DS Camera
<b>Max. resolution</b>	4096×3072	4096×3072 But large photos are reduced when loaded (see <b>Note 1</b> )		
<b>Min. resolution</b>	160×120 If photo is reduced, the result must satisfy this condition.			
<b>Data size</b>	JPEG 200 KB or less MPO 400 KB or less The picture quality is adjusted automatically inside library to fit within these restrictions (see <b>Note 3</b> ).	400 KB or less		10 MB or less
<b>Thumbnails</b>	Created automatically as 160×120 or smaller	1280×960 or smaller But reduced to 320×240 or smaller (see <b>Note 2</b> )		

**Note 1:** Images are shrunk to be no larger than 1024×768 for ImageDb and Nintendo 3DS Camera, or 320×240 for the Photo Selector Applet. Each image is shrunk to 1/2, 1/4, 1/8, or 1/16 of its original size, as appropriate.

**Note 2:** If there is no thumbnail, the image must satisfy the following conditions to be handled by ImageDb and the system: The image resolution must be 1024×768 or lower, and the size must be 200 KB or smaller for JPEG or 400 KB or smaller for MPO.

**Note 3:** Encoding is repeated while lowering the set quality until the image fits within the restriction. This feature is enabled by default, but it can also be disabled. For more information, see the *ImageDb API Reference*.

## 5.6 Image File Paths

Both ImageDb and Nintendo 3DS Camera save images under the DCIM folder on an SD Card.

- Filenames begin with HNI\_ followed by a four-digit number (between 0001 and 0100).

**Example:** HNI\_0123.JPG, HNI\_0123.MPO

- Folders (under the DCIM folder) begin with a three-digit number (100–999) followed by NIN03.

**Example:** DCIM/123NIN03/

Only JPG files can be saved as flat images. Each stereoscopic image is saved as both an MPO and a JPG file, each with the same name. It's easier to see the content of a stereoscopic image on a PC with the JPG file, but it is never displayed alone by the Photo Selector applet and cannot be handled by ImageDb.

You can also register and load managed image files other than those listed above as long as they conform to the following path conventions.

- Filenames begin with four alphanumeric characters and/or underscores followed by a four-digit number (0001–9999).

**Example:** ABCD0123.JPG, ABCD0123.MPO

- Folders (under the `DCIM` folder) begin with a three-digit number (100–999) followed by five alphanumeric characters and/or underscores.

**Example:** DCIM/123ABCDE

## 5.7 When Loading Images Other Than Photos Taken With the Nintendo 3DS Camera

---

Images other than photos taken with the Nintendo 3DS Camera can be loaded with the `ImageDb` library. Because of this, take the following measures so that the application will run properly even when loading images that are not 640×480.

- Enlarge, shrink, or trim the image so that it is displayed normally.
- Do not load images of a size that the application cannot handle.

## Revision History

Version	Revision Date	Action	Description
1.8	2015-03-20	Changes	<ul style="list-style-type: none"> <li>2.2 Available Pixel Resolutions for Images to Add</li> </ul> Revised notes about image pixel resolutions that can be added.
		Changes	<ul style="list-style-type: none"> <li>4.2 Message Examples</li> </ul> Added notes.
		Additions	<ul style="list-style-type: none"> <li>5.7 When Loading Images Other Than Photos Taken With the Nintendo 3DS Camera</li> </ul> Added precautions about images loaded.
1.7	2012-04-02	Additions	<ul style="list-style-type: none"> <li>5.3 Error Handling during Mounting</li> </ul>
		Changes	<ul style="list-style-type: none"> <li>2.3 Displaying Images in Nintendo 3DS Camera</li> </ul> Added note about display methods in the latest version.
		Changes	<ul style="list-style-type: none"> <li>5.1 Control Files</li> </ul> Added movies to description.
1.6	2011-11-25	Additions	<ul style="list-style-type: none"> <li>5.4 Restrictions on Handled Images</li> </ul> Added note specific to quality adjustment.
1.5	2011-11-07	Additions	<ul style="list-style-type: none"> <li>3 Using Loaded Images</li> <li>3.1 Focus Adjustment Values and Display Positions</li> </ul>
1.4	2011-08-10	Changes	<ul style="list-style-type: none"> <li>3.2 Message Examples</li> </ul> Added a note about checking for errors.
		Additions	<ul style="list-style-type: none"> <li>4.5 Image File Paths</li> </ul>
1.3	2011-07-12	Changes	<ul style="list-style-type: none"> <li>4.4 Restrictions on Handled Images</li> </ul>
1.2	2011-07-04	Changes	<ul style="list-style-type: none"> <li>2.3 Displaying Images in Nintendo 3DS Camera</li> <li>Added a note about MP files.</li> <li>3.2 Message Examples</li> <li>Revised examples to correctly reflect the error types</li> </ul>
		Additions	<ul style="list-style-type: none"> <li>4.4 Restrictions on Handled Images</li> </ul>
1.1	2011-04-12	Changes	<ul style="list-style-type: none"> <li>3.1 Errors That Require User Notification</li> <li>Made change in line with addition of result type</li> </ul>
		Additions	<ul style="list-style-type: none"> <li>4.3 Using Threads</li> </ul>
1.0	2011-02-02	Changes	<ul style="list-style-type: none"> <li>Changed document format to PDF.</li> <li>3.1 Errors That Require User Notification</li> <li>Changed error display from optional to required.</li> <li>4.1 Control Files</li> </ul> Clarified maximum number of files that can be managed.
		Deletions	<ul style="list-style-type: none"> <li>2.5 Controlling Distribution</li> </ul>
0.5	2010-12-30	Additions	<ul style="list-style-type: none"> <li>3 Error Handling</li> </ul>

Version	Revision Date	Action	Description
0.4	2010-12-15	Changes	<ul style="list-style-type: none"><li>• 2.3 Displaying Images in Nintendo 3DS Camera Reworked to clarify unclear explanations.</li><li>• 2.4 Saving Screenshots Added description of 3D images.</li></ul>
0.3	2009-12-04	Changes	<ul style="list-style-type: none"><li>• 4.1 Control Files</li></ul>
0.2	2009-12-01	Additions	<ul style="list-style-type: none"><li>• 4.2 Mounting SD Cards</li></ul>
0.1	2009-11-20	—	Initial version.

All company and product names in this document are the trademarks or registered trademarks of their respective companies.

© 2009–2015 Nintendo

The contents of this document cannot be duplicated, copied, reprinted, transferred, distributed, or loaned in whole or in part without the prior approval of Nintendo.