Mightex USB Camera TWAIN Application Guide

Mightex USB 2.0 color camera is mainly designed for microscopy and other scientific applications, in which cost-effective and ease of use are important. With USB 2.0 high speed interface and powerful PC software processing, the camera delivers excellent quality images at high frame rate. GUI application and SDK, as well as TWAIN driver are provided for user's application developments.

TWAIN Driver Installation

In the CD shipped with the Camera, there's a directory \TWAIN, it includes two sub-directories:
\MightexDS
\Documents

For install the TWAIN Driver to Windows, user can imply copy the \MightexDS (the whole directory with all its files) into "<WINDOWS>\twain_32", here <WINDOWS> is the directory where Windows is installed, for Windows XP, it usually is "C:\Windows". For Windows 2000, it's usually "C:\WINNT" (Here, I assume the Windows is installed on C: disk). So After installation, we have a directory like:

"<WINDOWS>\twain_32\MightexDS"

For uninstall the Mightex TWAIN driver, user can simply remove the "MightexDS" sub-directory.

TWAIN Specification Compliance

Mightex TWAIN driver is TWAIN 1.8 compliant, it supports the following TWAIN Operation Triplets:

DG_CONTROL:

DAT_IDENTITY
DAT_USERINTERFACE: ***Note: For Enable, Disable DS, UI can be (or not be) shown, while it's shown, it's always a modeless window.***
DAT_CAPABILITY
DAT_STATUS
DAT_PENDINGXFERS : ***Note: Currently, DS only supports single image transfer, so this operation always return ZERO count in cases of MSG_GET or MSG_ENDXFER. This also implies that application has to Grab and Transfer (if UI is shown) again for the next frame.***
DAT_SETUPMEMXFER
DAT_SETUPFILEXFER
DAT_XFERGROUP
DAT_DEVICEEVENT: ***Note: Currently, there's no event will be allowed to be generated by DS.***

DG_IMAGE:

DAT_IMAGEINFO
DAT_IMAGELAYOUT : ***Note: For camera, only several predefined frame sizes are supported, the user setting frame size will be changed to one of the predefined size (in a way DS thinks it's most proper), so it's recommended for application to MSG_GET the actual frame after a MSG_SET. So does the start position of the frame (Left,Top), camera will always sets it to even numbers.***
DAT_IMAGEMEMXFER
DAT_IMAGENATIVEXFER
DAT_IMAGEFILEXFER
DAT_PALETTE8 : ***Note : Currently, the camera will always generate 24bit RGB DIB, so there's actually no palette info can be get, it always return back empty palette information.<Even for Monochrome camera, the output image format is in 24bit RGB DIB as well>***
DAT_GRAYRESPONSE : ***Note: Currently, this operation is not supported. (Application always get TWRC_FAILURE for this operation)***
DAT_RGBRESPONSE : ***Note: Currently, this operation is not supported. (Application always get TWRC_FAILURE for this operation)***

DAT_CIECOLOR : ***Note: Currently, this operation is not supported. (Application always get TWRC_FAILURE for this operation)***

DAT_JPEGCOMPRESSION : Note: Currently, this operation is not supported. (Application always get TWRC_FAILURE for this operation), the DIB is NOT compressed.

CAPABILITIES:

****. Supported Capabilities:***

CAP_AUTHOR
CAP_INDICATORS ***: Note: It's always False as we don't have indicator for scanning process.***
CAP_UICONTROLLABLE
CAP_XFERCOUNT***: Note: We only support single image transfer, so it's always 1.***
CAP_DEVICEEVENT***: Note: No Device Events are supported for our camera.***
ICAP_XFERMECH: Note: ***We support all three types of transfer (native, memory and File).***
ICAP_IMAGEFILEFORMAT: ***Note: The camera only supports BMP format (TWFF_BMP).***
ICAP_PIXELTYPE: ***Note: We only support TWPT_RGB.***
ICAP_PLANARCHUNKY: ***Note: We only support Chunky.***
ICAP_BITDEPTH: ***Note: As the camera only generate 24bit DIB format image, it's always 24.***
ICAP_BITORDER: ***Note: It's always TWBO_MSBFIRST.***
ICAP_COMPRESSION: ***Note: Compression is NOT supported (TWCP_NONE)***
ICAP_BRIGHTNESS: ***Note: Our camera supports 0 to 10 (11 levels) brightness control, it actually controls the ISO parameters (Sensitivity of the camera)***.
ICAP_UNITS: ***Note: it's always TWUN_PIXELS.***
ICAP_XRESOLUTION: ***Note: it's always 1***.
ICAP_YRESOLUTION: ***Note: it's always 1.***
ICAP_PHYSICALHEIGHT: ***Note: For 1.3M camera, it's 1024 (pixels), for 3M camera, it's 1536 (pixels)***
ICAP_PHYSICALWIDTH: ***Note: For 1.3M camera, it's 1280 (pixels), for 3M camera, it's 2048 (pixels)***
ICAP_EXPOSURETIME: ***Note: We support 1—750 (ms) exposure time range.***

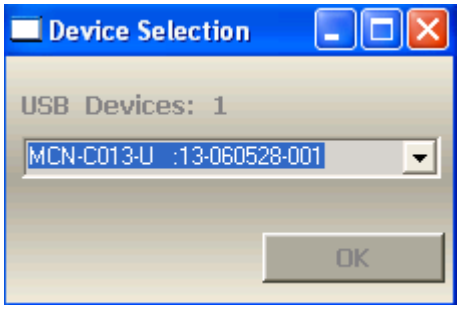
****. Extended Capabilities:***

CAP_XFERCOUNT
CAP_DEVICEEVENT
ICAP_XFERMECH
ICAP_IMAGEFILEFORMAT
ICAP_PIXELTYPE
ICAP_PLANARCHUNKY
ICAP_BITDEPTH
ICAP_BITORDER
ICAP_COMPRESSION
ICAP_BRIGHTNESS*
ICAP_UNITS
ICAP_XRESOLUTION
ICAP_YRESOLUTION
ICAP_EXPOSURETIME*

***Note:** Although these 14 Capabilities are listed as "Extended Capabilities", most of them are not configurable (they only support ONE particular value), only ICAP_BRIGHTNESS and ICAP_EXPOSURETIME are allowed to be configured.

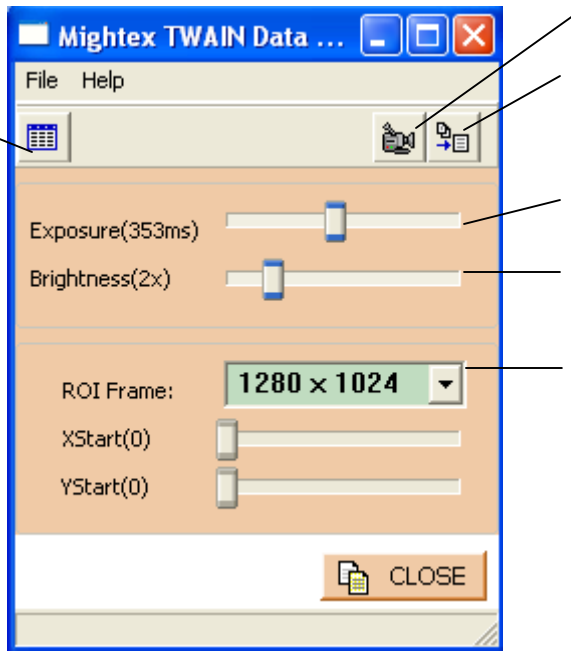
TWAIN Data Source User Interface

While the Data Source is opened (DG_CONTROL/DAT_IDENTITY/MSG_OPENDS), if there's more than one Mightex USB camera are connected to the PC, there's a "Device Selection" dialog will show up to let user to select the Camera application wants to control:



If there's only one camera connected, it won't be shown and the camera is opened automatically.

While Data Source is enabled (DG_CONTROL/DAT_USERINTERFACE/MSG_ENABLEDS), the Data Source UI may or may not show up (according to the applications ShowUI setting), the UI is as following:



Click it will show the preview window

Grab a frame

Transfer the image to Application

Exposure Time Control (1ms – 750ms), application may control it via ICAP_EXPOSURETIME

Brightness Control (0x – 10x), Application may control it via ICAP_BRIGHTNESS.

ROI Frame Control, only 32x32, 64x64, 160x120, 320x240, 640x480, 800x600, 1024x768, 1280x1024 Frame size are allowed for 1.3M camera, and two additional 1600x1200, 2048x1536 sizes are allowed for 3M Camera. Application may control it via DG_IMAGE/DAT_IMAGELAYOUT/MSG_SET.

Although it's recommended to use the DS with UI shown, application without UI is allowed, application can achieve this by setting ShowUI to False, this will hide the UI and application uses the programmatic ways to set Exposure Time and Brightness via capability negotiation, and then enable the DS, For the ROI, application might set it by DG_IMAGE/DAT_IMAGELAYOUT/MSG_SET in STATE4, that is after the DS is opened but before it's enabled (STATE5).