Video Playback on PHD70

April 15 2021

These scripts show how to implement the video clip playback on the PHD70 STD.

Video playback is enabled through a compiled Lua script *video_controller.lua*. This file must be included in the scripts folder in the Crank project. When the video playback is requested, the available memory is checked and if there is insufficient memory to allow for playback, the playback won't be enabled.

Functions

The functions included in the *video_controller.lua* script are:

```
-----------------------------------------------------------------------------
-------------------------------------------------------------------------

--  Description:

--    This function sets the rotation for video playback.

--

--  @function            [parent=#video_controller] set_rotation

--  @param               rotation A rotation_modes enumeration of the desired
rotation mode.

function video_controller.set_rotation(rotation)




-----------------------------------------------------------------------------
-------------------------------------------------------------------------

--  Description:

--    This function stops any running video playback.

--

--  @function             [parent=#video_controller] stop_video

function video_controller.stop_video()




-----------------------------------------------------------------------------
-------------------------------------------------------------------------

--  Description:

--    This function plays the provided video file. Any coordinates, if
supplied, must be specified relative to the native orientation of the
display,

--    even if the application itself will be rotated when running.

--
```

```
--    Any coordinates not provided in the call or given as nil will be set to the default values defined within the file.
--    Defaults: top = 103, left = 283, width = 480, height = 270
--      These defaults fit a particular Parker demonstration application. It is unlikely they will be suitable for a customer application.
--
--  @function             [parent=#video_controller] play_video
--  @param                videofile The fully-qualified name of the file to play.
--  @param                top       The screen position for the top of the video. Relative to native display orientation.
--  @param                left      The screen position for the left side of the video. Relative to native display orientation.
--  @param                width     The screen position for the width of the video. Relative to native display orientation.
--  @param                height    The screen position for the height of the video. Relative to native display orientation.
function video_controller.play_video(videofile, top, left, width, height)
```

Video Format

Videos must be in the follow format.

- Encoding must be in AVC (H.264) in an .mp4 container.
- Audio streams can be included but are not recommended to save space as the PHD has no audio playback capability.
- Size must be 720x480 pixels. Larger sizes waste space and smaller sizes will be scaled and appear pixelated during playback. It is recommended to used anamorphic encoding where the encoded frame is 720 pixels wide. This will expand to 800 pixels with minimal loss of video quality with much smaller file sizes.
- Playback must be between 25 and 30 fps.
- Video files must be included in the “videos” application directory or in a subdirectory (similar to the "images" subdirectory).
- Although there isn’t an explicit limit on the size of the video files, the overall size of the PHD Application package loaded into the PHD cannot exceed the limits specified for the PHD type.

To save time, a PHD70 encoding profile called **handbrake_phd70,json** for the video encoding tool, Handbrake, has been included as a reference. Simply import this file into your Handbrake project and then select the PHD70 profile from the Custom Presets.

Examples

In the Lua file where you wish to call the video playback functions, add this line near the top of your Lua scripts:

```
local video_controller = require("video_controller")
```

The functions can be called by issuing the following commands:

```
video_controller.play_video("/root/app/video/video1.mp4")
      -- this plays a video with default location and size.
```

Or

```
video_controller.stop_video()
      -- this stops any playing video
```

Also, various rotation modes are supported using the **video_controller.set_rotation_mode()** command. These modes are relative to the native orientation of the display.

The supported rotation modes are shown in the examples below using the **video_controller.set_rotation_mode ()** call

```
video_controller.set_rotation_mode( video_controller.rotation_modes.NONE )

video_controller.set_rotation_mode(
video_controller.rotation_modes.VERTICAL_FLIP )

video_controller.set_rotation_mode(
video_controller.rotation_modes.HORIZONTAL_FLIP )

video_controller.set_rotation_mode(
video_controller.rotation_modes.ROTATE_180 )

video_controller.set_rotation_mode(
video_controller.rotation_modes.ROTATE_90_RIGHT )

video_controller.set_rotation_mode(
video_controller.rotation_modes.ROTATE_90_RIGHT_VFLIP )

video_controller.set_rotation_mode(
video_controller.rotation_modes.ROTATE_90_LEFT )
```