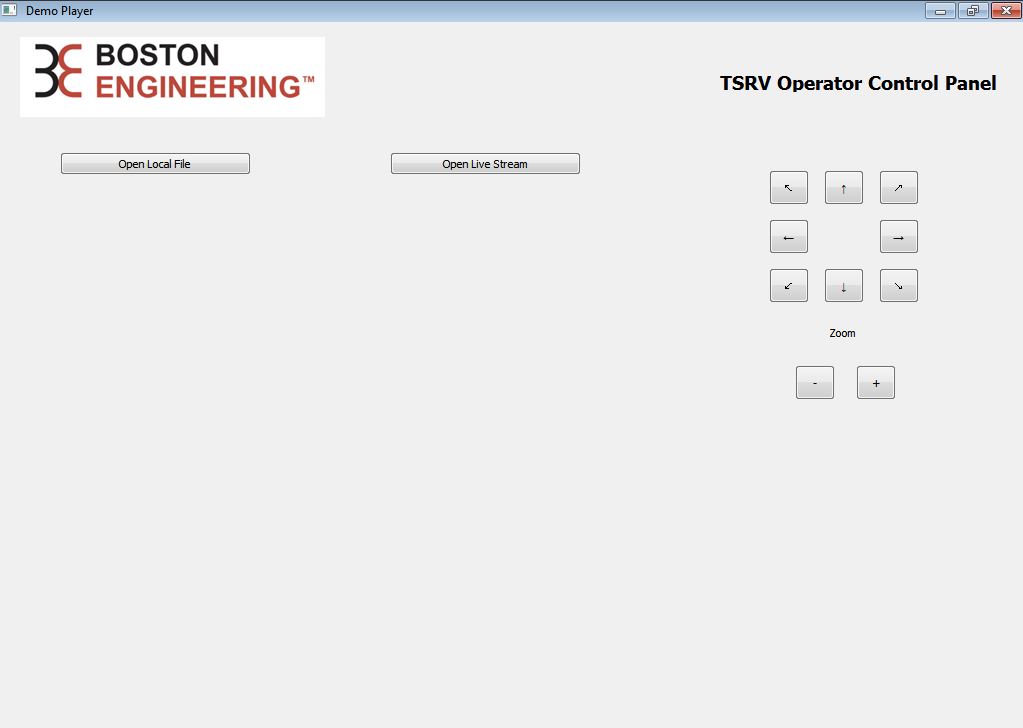
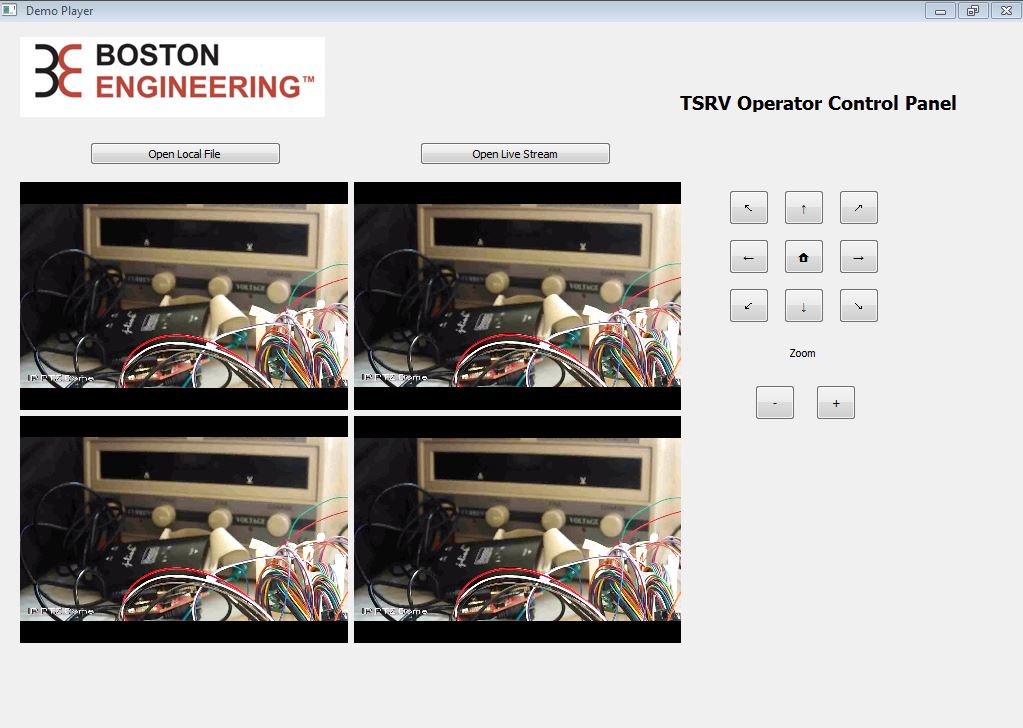
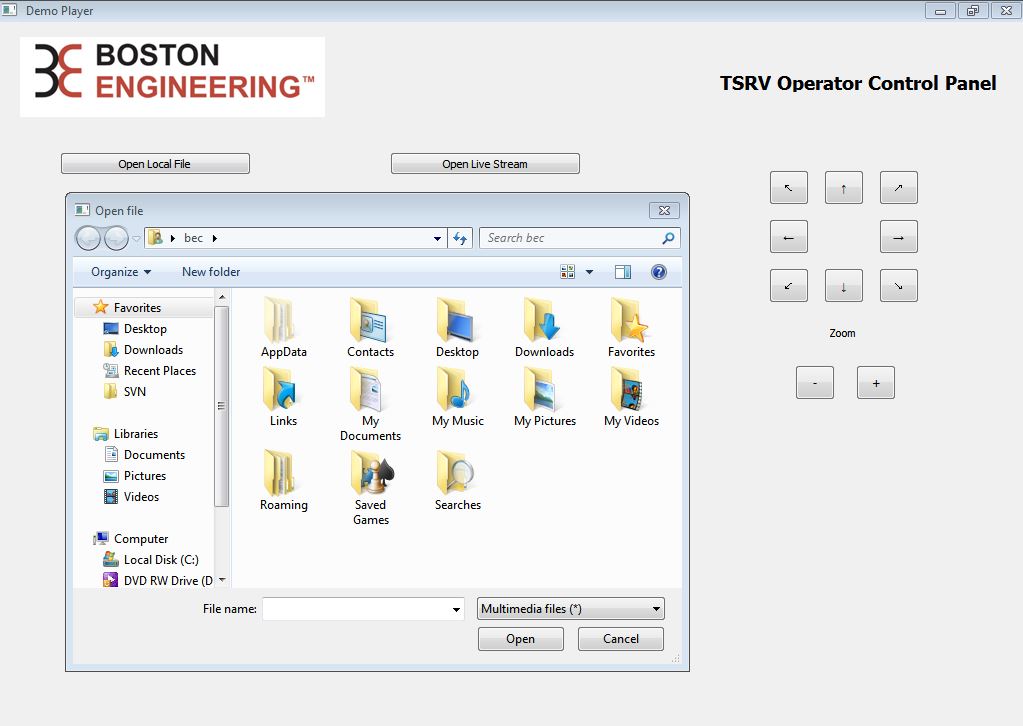
NRL001:

* Designed a user interface to control the Dahua SD22204T-GN IP Network PTZ camera using Qt (Qt Creator & Qt Designer)
* Interface has buttons for the operator to control the movement of the PTZ camera, as well as buttons for the camera’s zoom feature
* Interface allows user to play live stream from camera or to open a local file
* Development challenges:
  + Learning Qt
  + Displaying live stream
    - Used iSpy to find webpage URL used to communicate with camera
    - Tried to play video using built in Qt Multimedia Widgets to send a network request to the URL, but had troubles playing MJPEG live stream videos, was able to play static sample WMV and MP4 videos from web
    - Used VLC-Qt library to connect to libvlc instance and player, allowing for video live stream playback
      * Had a lot of build library path errors
        + Resolved by walking through compile output step by step
        + Modified project file to allow for custom library location by mapping file to correct libraries
        + Modified general project configuration to allow qmake to generate proper Makefile
      * Video delay (suspected due to VLC player)
        + No video delay when using Qt Multimedia Widgets to play MP4 and WMV videos from web, read threads of people trying to avoid using qt-VLC player because of video delay
        + IP camera video stream not delayed through iSpy
        + Not yet resolved: look into alternative ways to display IP camera live stream such as qx11EmbedWidget and qx11EmbedContainer or Phonon
  + Figuring out how to send commands to camera for PTZ movement
    - Found XML configuration script for PTZ control in iSpy GitHub repo
    - Used signals and slots to send HTTP GET requests to configuration script URLs to allow for camera movement
  + Designed UI interface using Qt Designer, and linked that to code through Qt Creator
  + Open local file feature
    - Found Qt classes to allow users to access directory structures and select local files
  + 4 Camera frames
    - Proves that there won’t be a video delay with more cameras playing
    - App may need to have different arrow keys so operator can move different cameras
    - Modify openUrl to map each video player to the different camera URLs, create new media instances

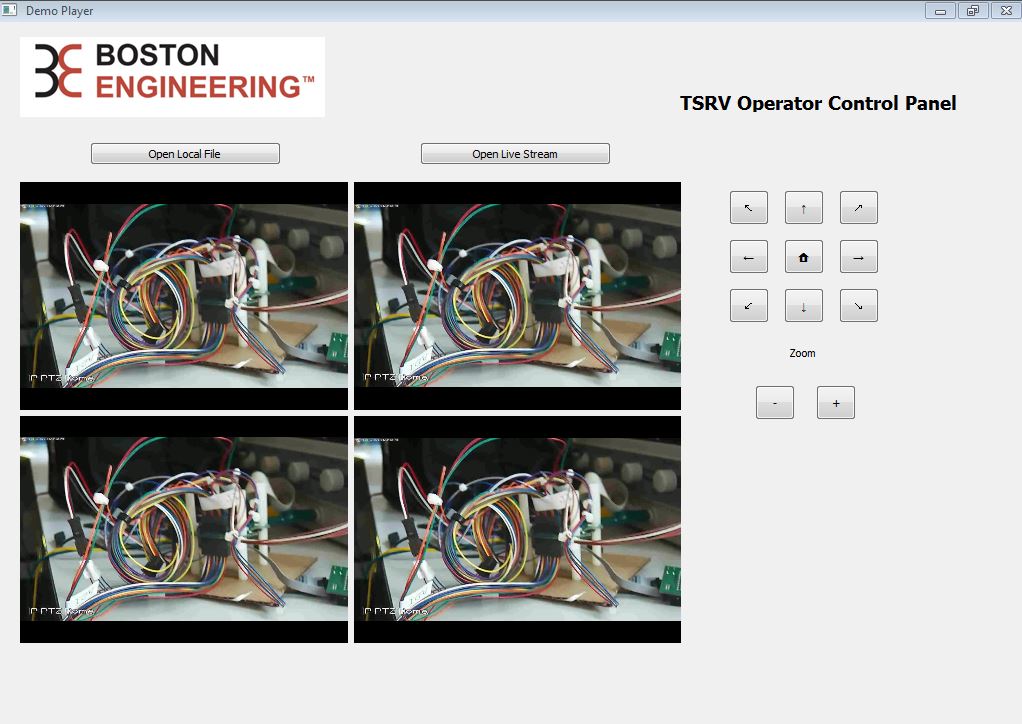


**Figure 1. TSRV Operator Control Panel Interface**

**Figure 2. TSRV Operator Control Panel showing live stream**



**Figure 3. Interface prompting user to choose local file to play**



**Figure 4. TSRV Operator Control Panel playing video recorded through iSpy**