Aupera Facial Recognition U30 Setup User Guide

Document Revision: 1.2

Aupera Technologies Inc.

Revision History

No.	Date	Description
0.1	Dec 08th, 2020	Created
1.0	Jan 15th, 2021	Re-organize
1.1	Jan 20th, 2021	Updates on section 2

Table of Content

Contents

Aupera Technologies Inc.

Aupera Technologies Inc.

1   INTRODUCTION

This document describes how to setup Face Recognition on Xilinx Alveo U30 board.

The demo system block diagram is as below:

The procedure is described as below:

```
┌──────────────────────────────────┐
│     U30 firmware installation    │
└──────────────────────────────────┘
                 │
                 ▼
┌──────────────────────────────────┐
│     Docker image installation    │
└──────────────────────────────────┘
                 │
                 ▼
┌──────────────────────────────────┐
│    Request license from App Store│
└──────────────────────────────────┘
                 │
                 ▼
┌──────────────────────────────────┐
│  Unzip client software on windows PC │
└──────────────────────────────────┘
                 │
                 ▼
┌──────────────────────────────────┐
│ Setup video streaming and perform test │
└──────────────────────────────────┘
```
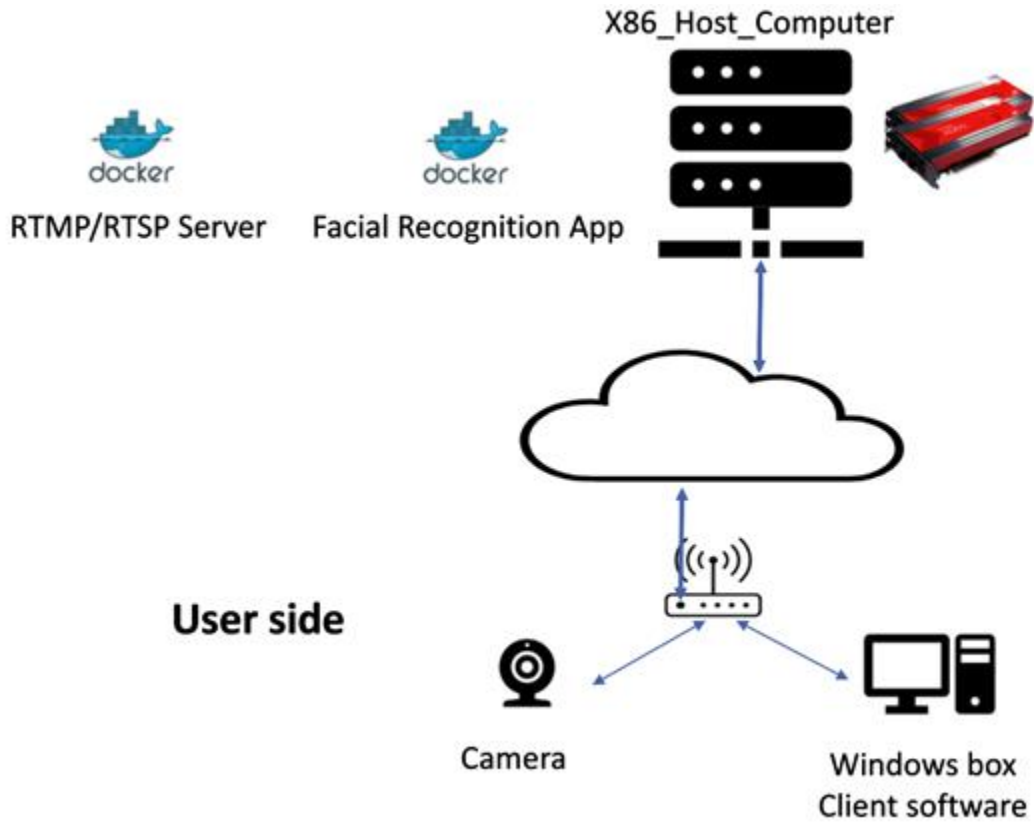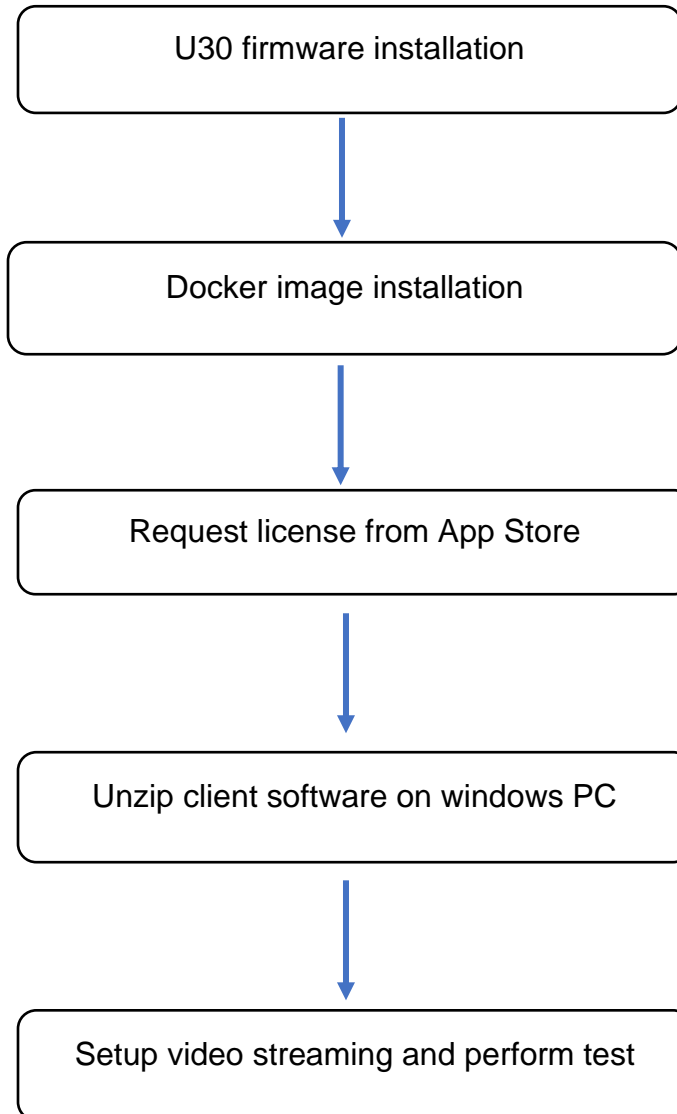
2   ALVEO U30 FIRMWARE INSTALLATION

Before Installation, please make sure that the server must support PCIe Bifurcation. Below are two examples of Bifurcation setup in BIOS.

SuperMicro:
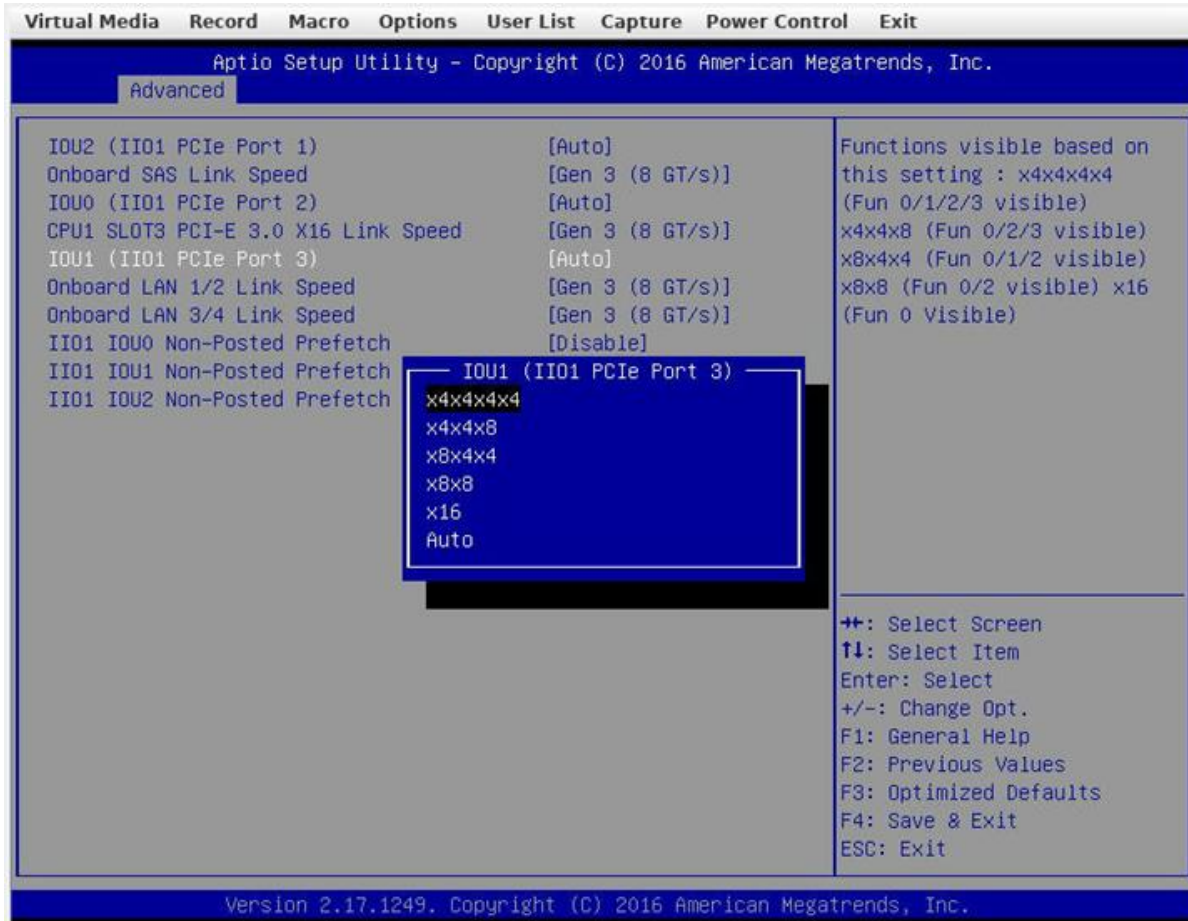
ASROCK:

There are two methods to flash the U30 board:

- flash over PCIe using XRT
- flash over JTAG

If this is the first time to flash, the JTAG option is required. The flash over PCIe option requires pre-flashed board.

The procedure is described as below chart:

```
            ┌──────────────────────────┐
            │     run xbutil list      │
            │ To find out XRT version  │
            │          on U30          │
            └──────────────────────────┘
                        │
                        ▼
        ┌────────────────────┐   No   ┌──────────────────────┐
        │  XRT ver <= 2.6.655?│──────▶│ uninstall current XRT,│
        └────────────────────┘        │    Install XRT2.6.0   │
                  │ Yes                └──────────────────────┘
                  ▼
        ┌──────────────────────────┐
        │  Xbmgmt utility to program│
        │     U30 bin file to flash │
        └──────────────────────────┘
                  │
                  ▼
        ┌──────────────────────────┐
        │ Cold reboot server which  │
        │       host U30            │
        └──────────────────────────┘
                  │
                  ▼
        ┌────────────────────┐   No   ┌──────────────────┐
        │ run lspci to check │──────▶ │   Tried 3 time?  │
        │ if U30 bootup      │        └──────────────────┘
        │ correctly          │
        └────────────────────┘
                  │                          │
          Yes, install docker        Yes, JTAG program flash
```
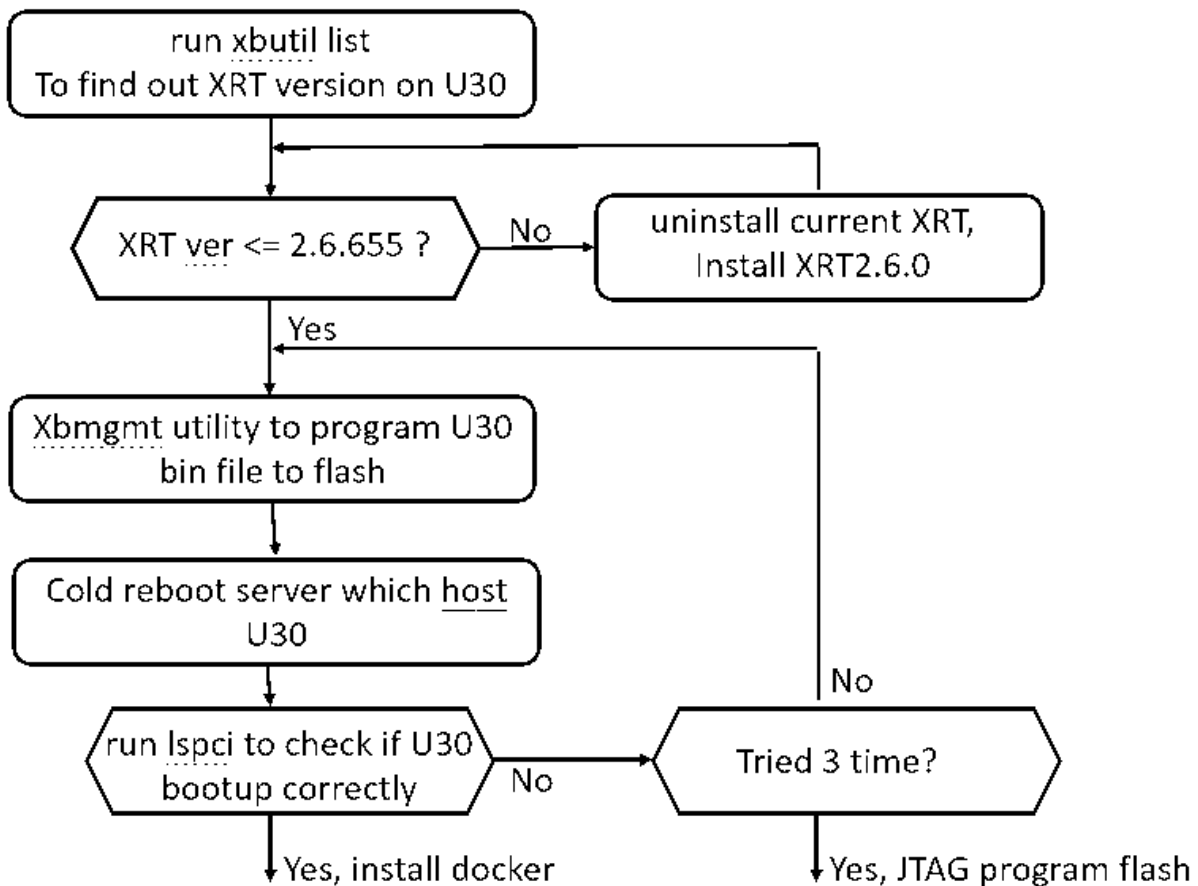
2.1   Firmware Installation using XRT over PCIe

2.1.1 Prerequisites

1. Aupera U30 firmware QSPI flash dump file. The firmware file is named as xu30-qspi-burn<version and datecode>.bin.
2. One U30 board with two xcu30 devices
3. One x86 Ubuntu 18.04 machine supports PCIe x8 bifurcation with Xilinx XRT installed.

2.1.2 Procedure

1. Source XRT env and check the current XRT version:

   $cd /opt/xilinx/xrt/

   $source setup.sh

$ xbutil --version

> XCLMGMT: 2.6.655

If the version later that 2.6.655, remove the current XRT in directory /opt/xilinx/xrt and install older version 2.6.0

2. Read the PCIe bus of the board:

$sudo lspci -d 10ee:

It will show something like:

> 07:00.0 Processing accelerators: Xilinx Corporation Device 503d (rev 02)
>
> 07:00.1 Processing accelerators: Xilinx Corporation Device 503c (rev 02)
>
> 08:00.0 Processing accelerators: Xilinx Corporation Device 503d (rev 02)
>
> 08:00.1 Processing accelerators: Xilinx Corporation Device 503c (rev 02)

The BDF IDs, 07:00.1 and 08:00.1, are the card_id used in the next step. The U30 board has 2 FPGA devices (PCIe endpoints)

3. Flash the U30 board using XRT xbmgmt utility:

$sudo /opt/xilinx/xrt/bin/xbmgmt flash --shell --card <card_id> --path <binfile>.bin

where <card_id> is the ID read from the step 2, like 07:00.1, and <binfile> is the file name of the Aupera firmware QSPI flash dump file.

It will show the process as below:

> Idcode byte[0]=20
>
> Idcode byte[1]=bb
>
> Idcode byte[2]=21
>
> Idcode byte[3]=10
>
> Idcode byte[4]=44
>
> Idcode byte[5]=0
>
> Erasing flash............................
>
> Programming flash........................
>
> Verifying........................
>
> Shell is updated successfully
>
> Cold reboot machine to load new shell on card

After completion, flash another one with the second card_id (like 08:00:1) read in the step 2 and the same flash dump file.

Note: The current firmware supports XRT version 2020.1.2.6.0 or 2020.1.2.6.655. The new firmware which supports later version XRT is expected to be available in Q2 2021.

The XRT 202010.2.6.0 deb files are included in the package.

4. Power down and power up (cold reboot) the host to update the FPGA image from flash.

Go to the section 3 for **the driver and docker Installation**

2.2 Firmware Installation over JTAG

This method is used for the first time programing the flash or there are any issues that XRT over PCIe doesn't work.

2.2.1 Prerequisites

1. Aupera U30 firmware QSPI flash dump file and Zynq FSBL file.
2. One U30 board with two xcu30 devices
3. One x86 Ubuntu 18.04 machine supports PCIe x8 bifurcation, this document will call this computer as the "X86_Host computer"
4. One Xillinx Alveo Programming (JTAG) Cable (Please refer to https://www.xilinx.com/support/documentation/boards_and_kits/accelerator-cards/ug1377-alveo-programming-cable-user-guide.pdf)
5. One x86 Ubuntu computer with Vivado or Vivado_Lab 2019.2 installed; this document will call this device as the "JTAG computer"

```
        ┌─────────────────────────┐
        │     JTAG computer       │
        │  with Vivado installed  │
        └─────────────────────────┘
                    │
                    │ Micro USB
                    ▼
        ┌─────────────────────────┐
        │     Xilinx Alveo        │
        │ Programing (jtag) cable │
        └─────────────────────────┘
                    │
                    │ Ribbon cable
                    ▼
        ┌─────────────────────────┐
        │   ┌─────────────────┐   │
        │   │ U30 board on PCIE│  │
        │   └─────────────────┘   │
        │  Motherboard X86 computer│
        └─────────────────────────┘
```
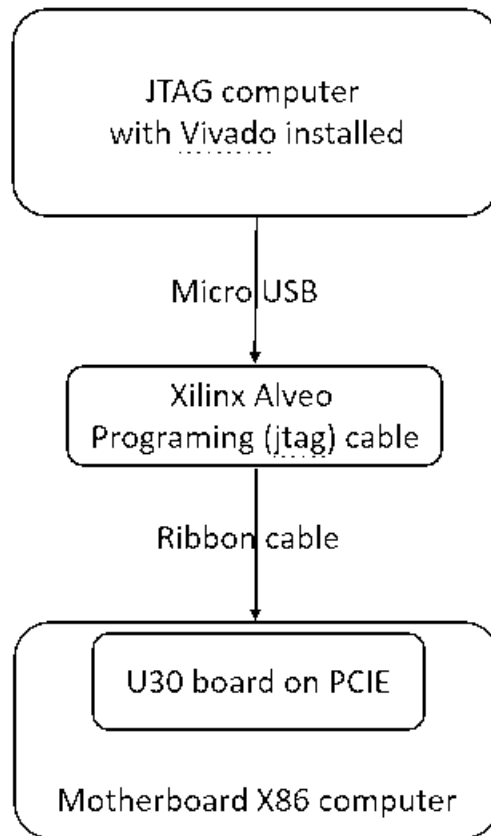
2.2.2 Procedure

1. Connect the Alveo Programming Cable to the U30 board, as shown in the diagram above.
2. Keep the X86_Host computer off, only Power on the JTAG computer.
3. On the JTAG computer, run `sudo minicom` to connect to the U30 with baud rate set to `115200 8N1`. There should be two terminals connected to each of xcu30 devices.(start minicom terminal for each /dev/ttyUSB, if you not sure which are xcu30 serial console)
4. Power on the X86_Host computer, which will provide power to both the U30 board and the X86_Host computer.
5. At the same time of last step, keep pressing the required key (see instructions from the terminal, usually pressing `s` would work) from both the minicom terminals on the programming computer to stop the autoboot to put the Linux system on both xcu30 devices to the halt state (U-boot ZynqMP prompt). The time window for doing this is short, when missing, please unplug the X86_Host computer's power, wait for about 1 minute and do step 3 again.

Note: ignore this step if this is the first-time flash programming(U30 flash chip empty).

6. From the JTAG computer, open Vivado or Vivado_Lab and do:
7. Click on "Open Hardware Manager", then "Open target", then "Auto Connect", then we should see the xcu30 devices listed as:
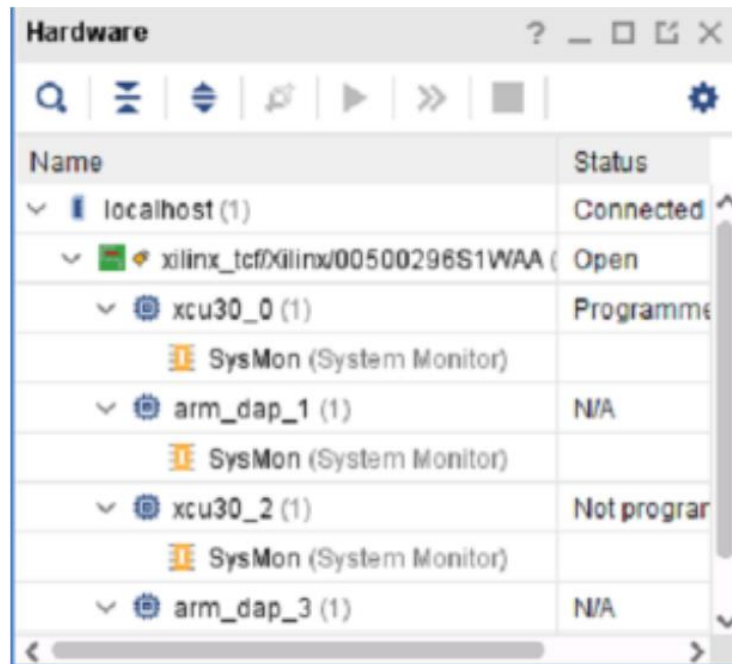

    a. For each "xcu30", right click on them and select "Add Configuration Memory Device...".

    b. On the popup window, use the following configurations:
        i. Ensure Device is "xcu30_<number>"
        ii. For Filter, select "Micron" for Manufacturer, "qspi" for Type, "1024" for Density (Mb), "x2-single" for Width
        iii. Select Configuration Memory Part with alias of 1.8v
        iv. Press "OK"

    c. On the next popup window:
        i. select Configuration file as the .bin file Aupera provided
        ii. select Zynq FBSL as the .elf file Aupera provided
        iii. Press "OK"

    d. Then Vivado/Vivado_Lab should start the installation process, when finished we should see:
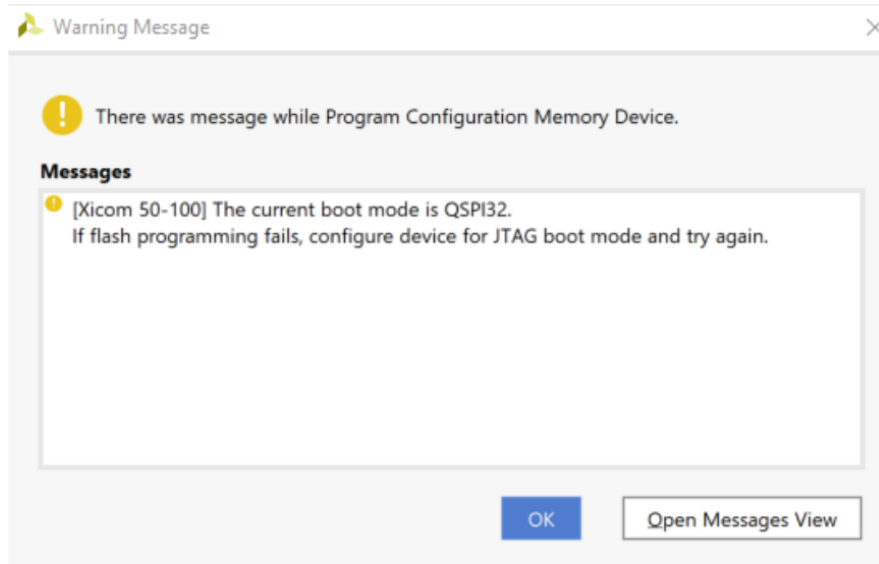
and
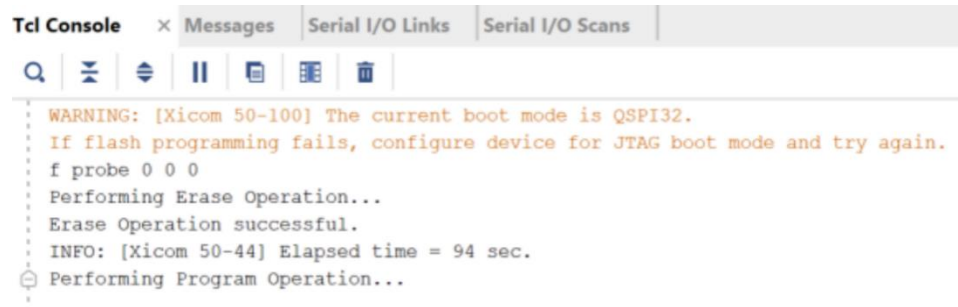
    e. Make sure we do the firmware installation for both of the U30 devices.

8. Disconnect the Alveo Programming Cable.

2.3 Restore to Xilinx XRT firmware

2.3.1 Prerequisites

1. Xilinx U30 firmware QSPI flash dump file. For example: file BOOT_full_RC5_5.bin from Xilinx.
2. One U30 board with two xcu30 devices
3. One x86 Ubuntu 18.04 machine supports PCIe x8 bifurcation and running Aupera U30 firmware.

2.3.2 Procedure

1. Split Xilinx QSPI flash dump file into 2 parts, first 16MB as mtd0.bin, and others after 16MB as mtd1.bin:

    $dd if= BOOT_full_RC5_5.bin of=mtd0.bin bs=1M count=16

     $dd if= BOOT_full_RC5_5.bin of=mtd1.bin bs=1M skip=16

2. Transfer mtd0.bin and mtd1.bin to U30 card by scp, the password of U30 is "VideoAlive@2020"
    $scp mtd0.bin mtd1.bin root@10.10.10.2:/tmp

3. login with ssh or serial console, program mtd0.bin and mtd1.bin into U30 card QSPI flash:
    $flashcp -v mtd0.bin /dev/mtd0

    $flashcp -v mtd1.bin /dev/mtd1

After completion, flash another one with the second card_ip address (for example 10.10.10.6) read in the step 2 and the same flash dump file.

Power down and power up (cold reboot) the host to update the FPGA image from flash.

Note: you can also program Xilinx XRT firmware to U30 board over JTAG cable.

2.4 Firmware upgrade

2.4.1 Prerequisites

U30 board running Aupera firmware

2.4.2 Procedure

The same produce as the section 2.3.2, but use a new version of Aupera QSPI flash dump file.

3   U30 DRIVER AND DOCKER INSTALLATION

3.1   Prerequisites

1) Aupera release package includes files: aupera_face_recognition_<version>.tar, docker_run.sh, aupera_drivers.tar.gz
2) One U30 board with two xcu30 devices
3) One x86 Ubuntu 18.04 machine supports one PCIe x16 for U50 and one PCIe x8 as dual x4 bifurcation for U30, this document will call this computer as the "X86_Host computer"
4) Internet access for the X86_Host computer

3.2   Procedure

1. Prepare essential software in X86_Host computer:
   1) $sudo apt update;sudo apt install make build-essential nfs-kernel-server docker docker-containerd docker.io
   2) Create directory /opt/aupera/face-recognition (directory /opt/aupera/face-recogniton is <NFS_ABS_PATH> used in the next steps), change its owner to your <UID>:<GID>, add nfs share directory in /etc/exports for U30 FPGA read/save files:
      /opt/aupera/face-recognition *(rw,sync,no_root_squash)
   3) Restart some nfs related services:
      $sudo service rpcbind restart
      $sudo service nfs-kernel-server restart
   4) Create directory <NFS_ABS_PATH>/drm/, then please refer to the section 5 to generate a license file (cred.json), and copy it to <NFS_ABS_PATH>/drm.
   5) Get and load Aupera modified XDMA kernel driver, a tar package named aupera_driver.tar.gz, (rebuild them to fit your system for the first time installation):
      $tar -xzvf aupera_driver.tar.gz
      $cd aupera_driver
      $sudo ./install.sh

      Note: if installation fails, please first remove xocl by running

      $sudo rmmod xocl

   6) Confirm kernel modules had loaded
      $lsmod | grep xdma
   7) Get Aupera docker file aupera_face_recognition_1.0.1.tar and run script docker_run.sh:
      $docker load -i aupera_face_recognition_1.0.1.tar
      $./docker_run.sh <REPOSITORY>:<TAG> <CONTAINER_NAME> <NFS_ABS_PATH>

Here <REPOSITORY>:<TAG> is the repository name, like aupera_face_recognition:1.0.1, which can get from command 'docker images', <CONTAINER_NAME> is a user defined container name, like face, and <NFS_ABS_PATH> is the nfs share directory path added to /etc/exports (described in section 3.2.1.2), like /opt/aupera/face-recognition.

an example of the command line:

$./docker_run.sh aupera_face_recognition:1.0.1 face /opt/aupera/face-recognition

8) Driver, docker and software installation complete, if you restart X86_Host computer, most of the steps in this section can bypass, the procedure is as follows:

First, install aupera xdma drive in section 3.2.1.5;

Second, start docker by command below:

$docker container start <CONTAINER_NAME>

Finally, exec docker by command below:

$docker container exec -it <CONTAINER_NAME> bash /etc/init.d/start.sh.

4   U30 FACIAL RECOGNITION LICENSE SET UP

1. Xilinx Appstore account
- Setup a Xilinx Appstore account (https://appstore.xilinx.com/ -> Manage Account).
- Purchase the Aupera Facial Recognition product from the Xilinx Appstore, a free trial is also available. The product can be purchased with a floating or node-locked license. Please note: Each U30 board contains 2 devices and therefore consumes 2 nodes to be fully utilized.
- At least one cred.json file (access key) must be created for your account (Manage Account -> Access Key -> Create an Access Key -> Download JSON). This file identifies your account to the Appstore during runtime and must be placed in the directory: <NFS_ABS_PATH>/drm/cred.json.

2. Node-Locked License
- To use a node-locked license purchased from the store, the U30 must be connected to the internet during runtime to generate a license file. After the license file is generated for the first time, it can unlock the DRM protection while offline and will only work for the specific FPGA's DNA it was generated for.
- To use node-locked licensing, the option: "nodelocked": true, must be set in the conf.json file located at <NFS_ABS_PATH>/drm/conf.json.

3. Floating License / Free Trial
- A floating license purchased from the store can be used immediately. A floating license cannot be used while offline.
- To use floating licensing, the option: "nodelocked": false, must be set in the conf.json file located at <NFS_ABS_PATH>/drm/conf.json.

5   CLIENT SOFTWARE INSTALLATION AND TEST EXECUTION

The current client software only supports windows, no installation is required, after de-compression, double-click dist/app/app.exe to execute the client.
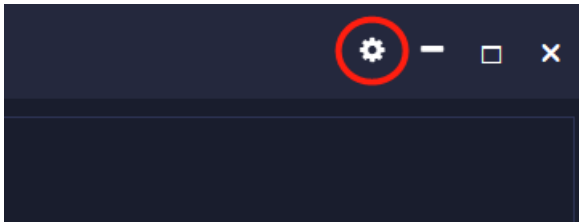
5.1 Login

Open Client (app.exe), you will see the sign- in page.

The default username and password: admin/admin123.

5.2 System configuration

Click the settings button on the upper right to enter the system configuration pop-up window. The configuration includes:

1. Device connection configuration.

2. Client language configuration.

5.2.1 Connecting X86_Host_Computer

1. Open the client and click the configuration button on the upper right to enter the system configuration.

2. Modify the device address to connect to the AI device. The address format is http://device-ip:56108, where device-ip is the device IP address, the port number is 56108;

3. Click the "OK" button to save the submission, and then restart the client software.

4. After the client logs in, it will automatically connect to the device. If the connection is successful, the software version information will be displayed, as shown in the figure below. If the connection fails, you cannot get relevant software version information. At this time, you need to check whether the device address is correct and valid.

Aupera Technologies Inc.

5.2.2 Language options

The client supports English and Chinese, and the default is English.

5.3 Video Preview

After successfully logging in to the server, the "Video Preview" module is displayed by default. The functions include:

1. Video stream management

2. Video preview

### 5.3.1 Video stream management

5.3.1.1 Add video streams

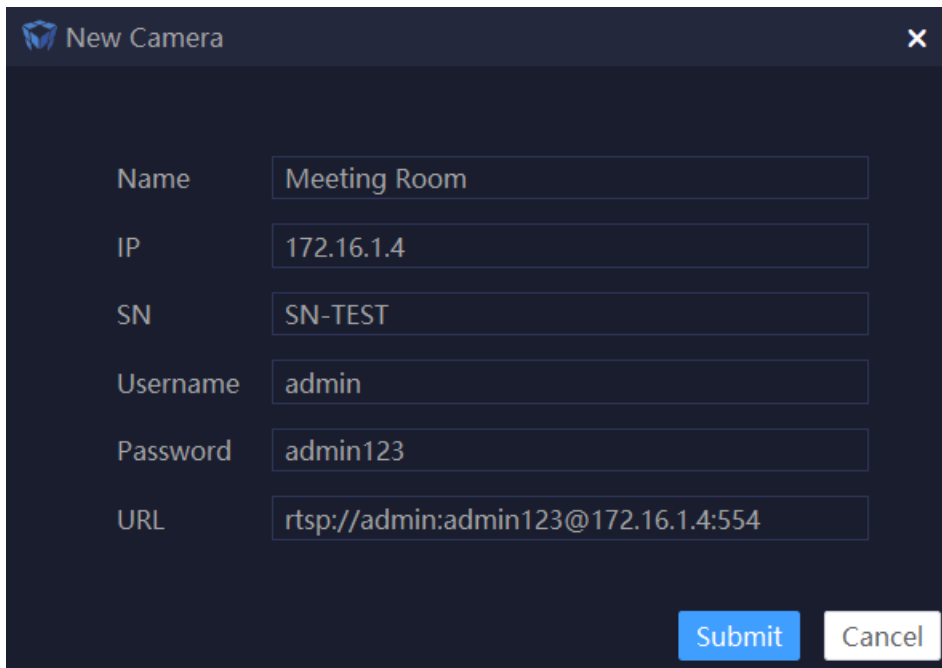There are two ways to add a video stream, steps as follows:

1. ONVIF device search

The device search uses ONVIF protocol to look for cameras on the local network. Click the 'ONVIF search device' on the upper left button to start the search task.

Note: If you need to preview the device searched by ONVIF, you need to configure a password for it. Click the configuration button to the right of the device name, enter the username and password of the camera, and then click the "Submit" button.
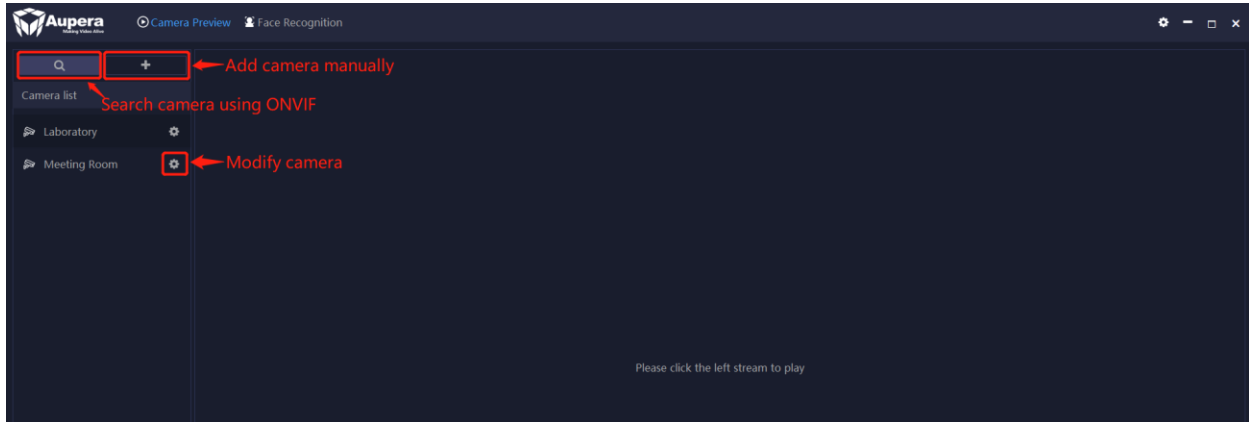
2. Manually add video stream

For a known video stream address, you can click the "+" button above the "Camera List" to manually configure and add, the configuration window is as shown below:

1) Name: the name of the camera

2) IP: Camera IP address, please ensure the uniqueness of each camera IP address in the list.

3) SN: Device serial number item, it supports combination of numbers and letters, please ensure the uniqueness of each camera SN in the list.

4) Username and password: the username and password of the camera, if not, you can fill in the default value admin/admin.

5) URL: The video stream address of the camera. Please ensure that the address is available and accurate.

5.3.1.2 Delete or edit IP cameras

After adding a camera, click the configuration icon button on the right side of the target camera. At this time, the configuration information of the camera can be modified, or click the "Delete" button below to delete the device.

5.3.2 Video preview

After adding the camera, you can click to select the camera to preview.

Click the "x" icon button at the bottom right of the play area to close the play preview.

5.4 Facial Recognition application

Facial recognition module includes face gallery management and task execution.

5.4.1 Gallery management

Gallery management includes face library management (creating and deleting different face libraries) and gallery manager (adding and deleting face IDs)

5.4.1.1 Face library management

Click the "Face Recognition" button in the navigation bar of the client, and then click the "Gallery Management" button on the right to enter the gallery management module.

On the left is the face library list, where you can add, configure and delete different face libraries, for example, we can create a new "whitelist" face library and "blacklist" face library.

1. Adding face library

Click the "+" button on the right side of the face library list in the figure above to create a new face library. At this time, as shown in the figure below, fill in the name of the face database in the pop-up box, such as "BlackList".

Note: The name of the newly created face library cannot be the same as the name of an existing face library, otherwise it will fail to create this library.

2. Delete the face library

Click the "Delete" button to the right of the corresponding face library name to delete the face library.

Note: This operation will affect the currently running task that uses the face library, and will also cause the detection record of the face library to display differently.

5.4.1.2 Gallery manager

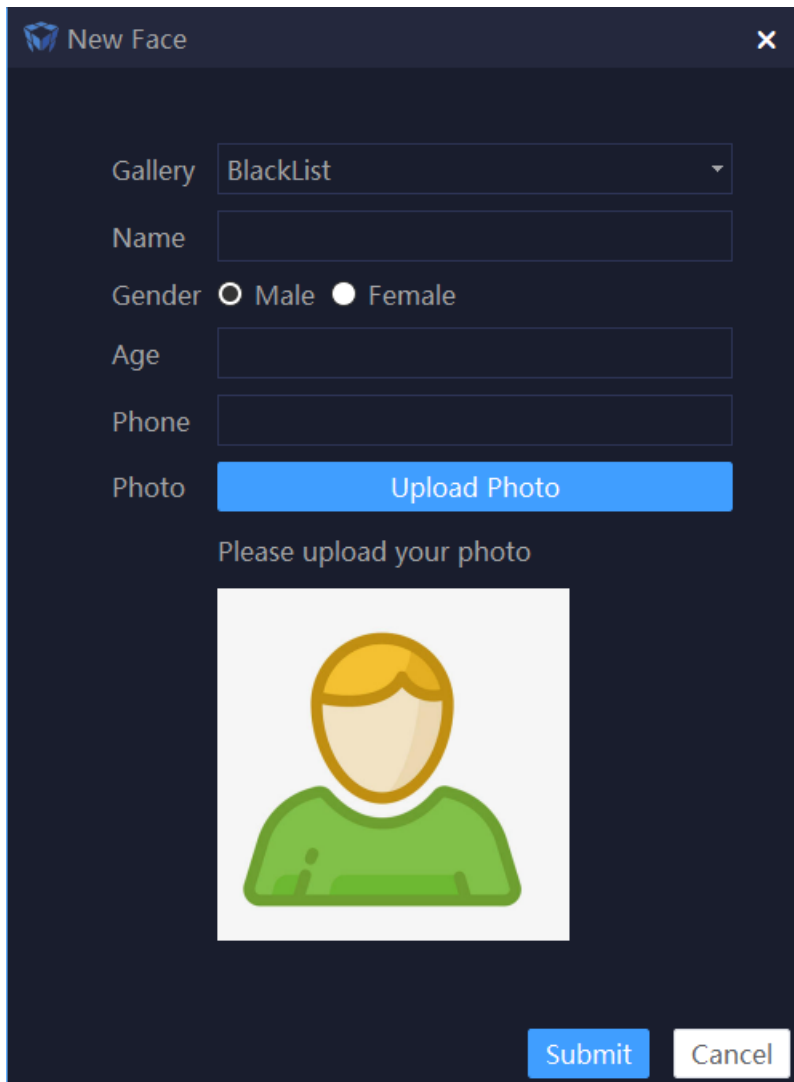Before adding face IDs, the face IDs database should be created first.

1. Adding a new face ID

    Click the 'New face' button to load new face ID.

    As shown in the figure below, you can select the target face database in the "face database" item on the left, and fill in the name (required), gender, age, phone number, and the last upload face ID photo.

    The system will test and rate the uploaded photos, and display "photo qualified" if the uploaded ID meets the specifications, otherwise please replace the photo.

After adding a face, you can click to view the face list under the corresponding face library. If there is no immediate refresh, you can click the "Refresh Face" button above to refresh the list of faces.
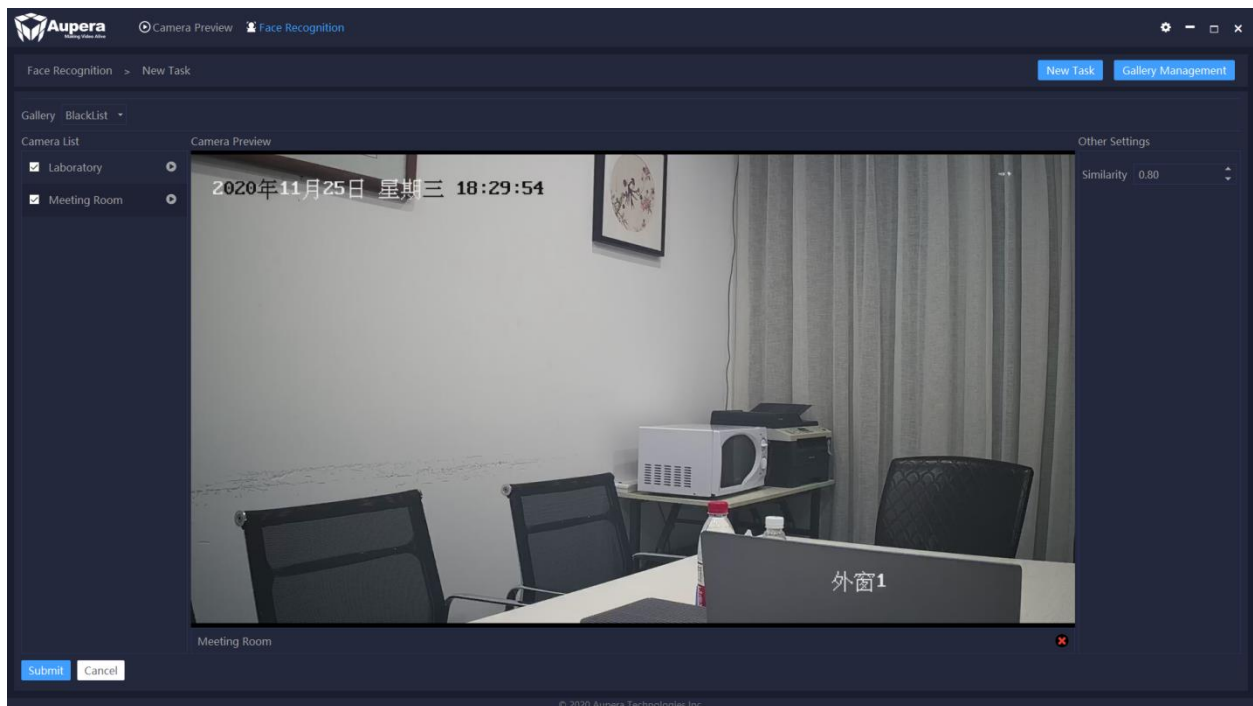
2. Delete face ID

You can find the face that needs to be deleted by clicking to open the corresponding face library to preview the face list, and click the "X" button at the top right of the face record to enter row to delete.

5.4.2 Task execution

There are 3 steps:

1. Create a new task

2. Task list

3. Task monitoring.

5.4.2.1 Create a new task

1. The basic operation process: select face library --> select camera --> adjust similarity --> submit task.

2. Parameters that need to be configured when creating a new task:

3. Face library: click on the top left to select in the drop-down box

4. Camera: Click to check in the camera list on the left and support playback and preview

5. Similarity: You can click the up and down buttons to add or subtract or directly click to enter the number to modify, the effective value range is between 0-1. The larger the value, the higher the recognition accuracy. But it will affect the recall rate. The recommended value range is 0.7~0.85, which can be adjusted according to the actual situation.

Note: In one submission task, one "face library" can only correspond to one "similarity" rate, but it can be used by multiple camera tasks; one camera task can select multiple "face libraries" at the same time. After submitting the task, it will automatically jump to the "Task List" panel, and you can view the task at this time to check relevant parameters.

5.4.2.2 Task list

After creating a new task, click to enter the "Face Recognition" function module, the task list and task details are displayed by default, as shown below:
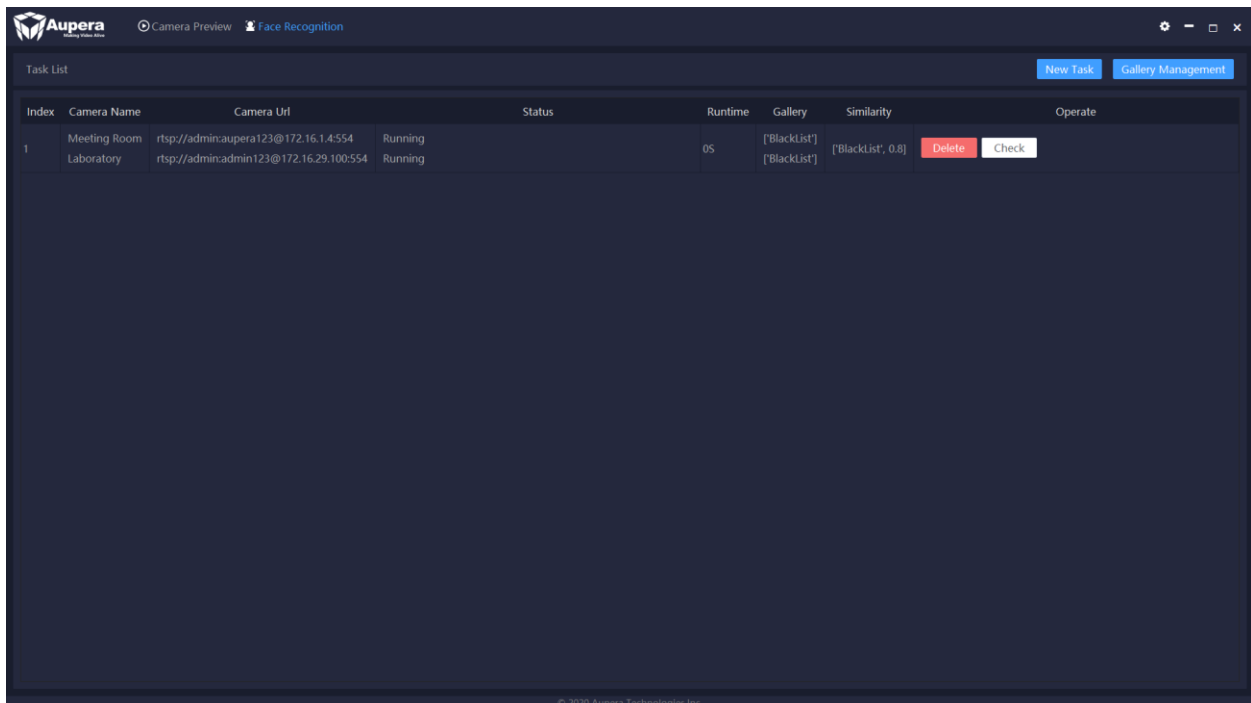
1) Sequence number: The time sequence of the task creation, the newly created task sequence number is bigger.

2) Camera name: the name of the camera selected when creating the task.

3) Camera address: the video stream address of the camera used by the task.

4) Status: The running status of the task, there are mainly two types: "running" and "abnormal". The occurrence of "abnormal" means that the task has stopped running.

5) Face library: The face library used by the task, a camera video streaming task can use multiple face libraries at the same time, and the people used by different video streaming tasks.

6) The face database is separated into different lists.

7) Similarity: The recognition threshold used by the task. Faces with a similarity greater than this value will be recognized and saved during detection.

8) Operation: There are two functions of delete and view. The "Delete" button is to delete the task, and the "View" button is to enter the "Task Monitoring" panel.

5.4.2.3 Task monitoring

Task monitoring is mainly for viewing the detection and recognition result records of face recognition tasks, including two functions:

1) View the alarm details.
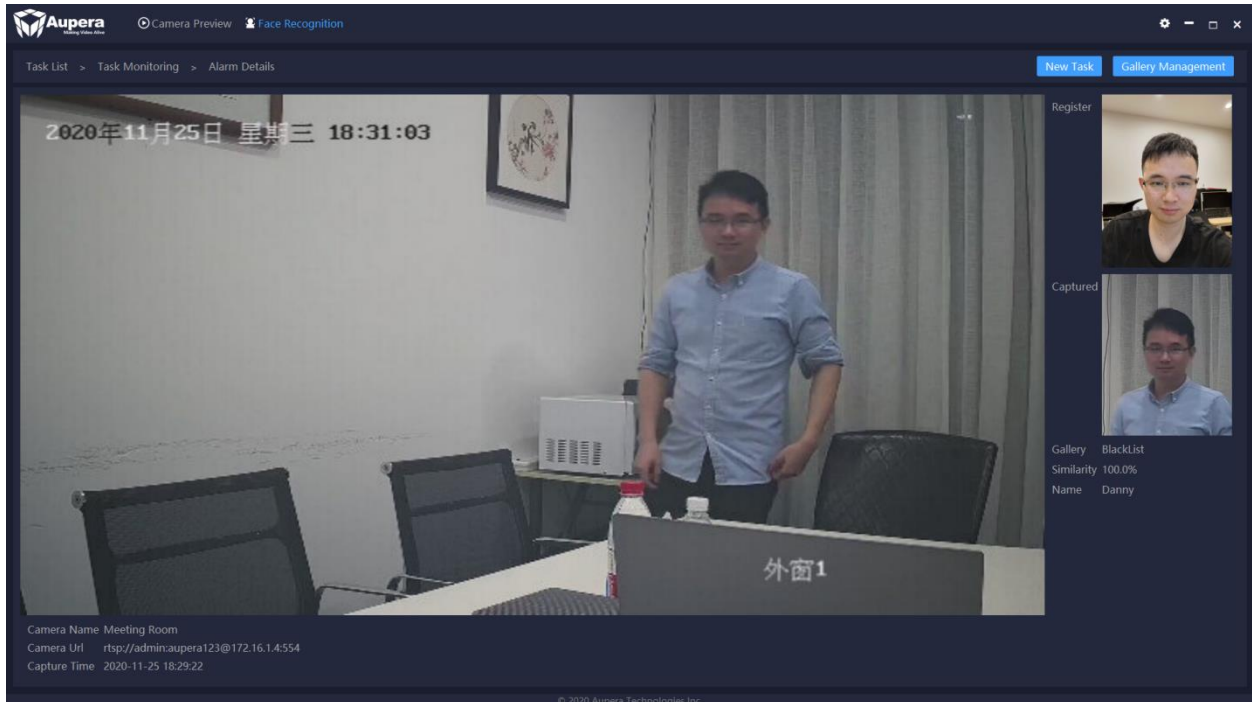
2) Search and filter alarms.

1. View alarm details:

This panel can view all the detection results of the current task. It is not automatically refreshed by default. You can click the "Update" button to get the latest results.

The test records are arranged according to time, with the latest test results at the top.
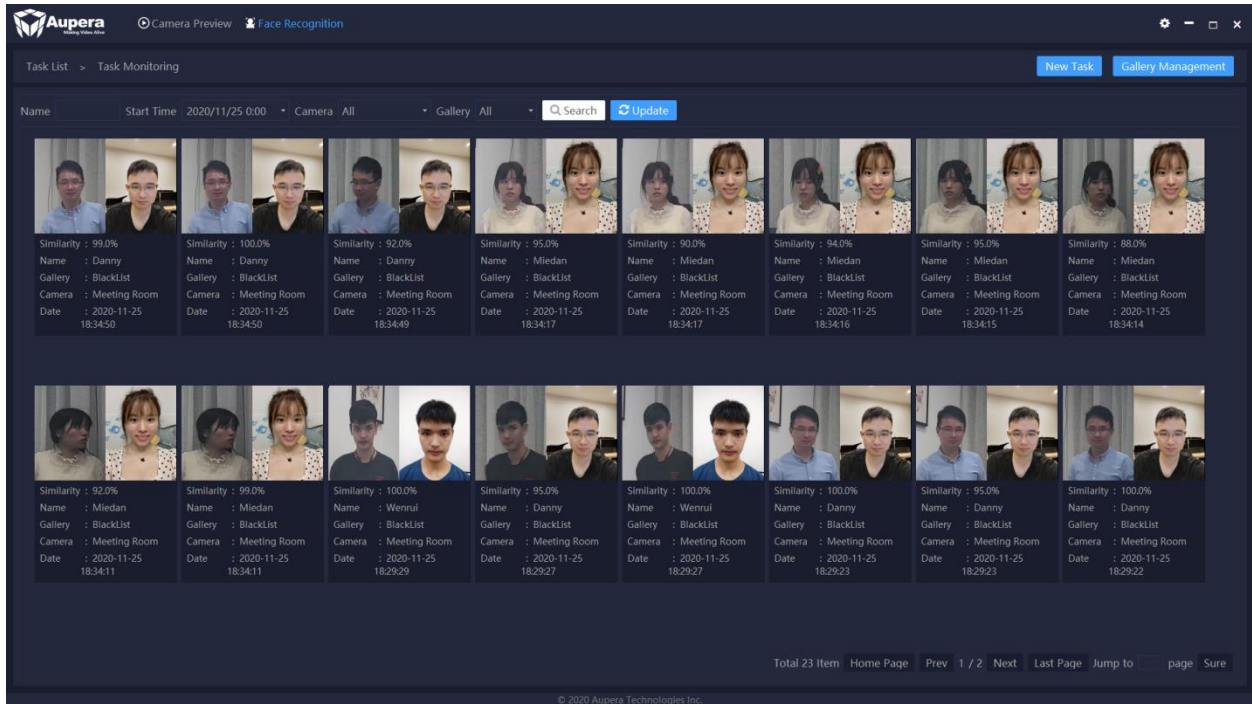
"Alarm details" view, you can double-click a single detection record to enter the "Alarm Details" panel to view the detailed information of the detection record, as shown below.
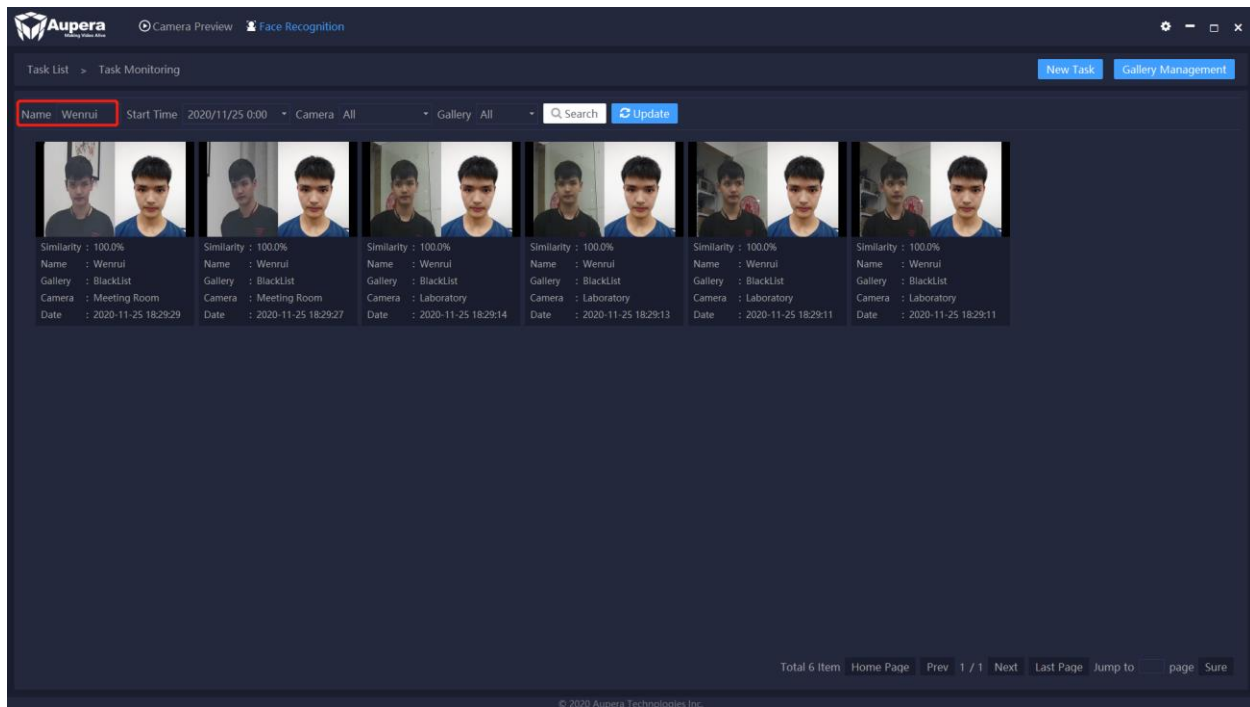
2. Search and filter alarms:

1) Support name search and partial name search

2) Support time stamp search

3) Support selecting designated camera and face library search.

After inputting the search criteria, click the "Search" button to obtain the test results, as shown in the figure below:

5.5 Frequent Q&A

1. The system misreports the Trojan virus

If you find a security warning during installation and use, you can add the dist/app/app.exe program executable file to the system whitelist.

If it is deleted by mistake by system quarantine, take Win10 as an example, you need to go to Windows Security Center --> Virus and Threat Protection --> Protect History,

Find the dist/app/app.exe executable file of the program to restore.

2. Point-to-point mode

The current client uses a point-to-point method to directly connect to the AI device, and the detection result is also directly transmitted to the client by the AI device, so multiple clients connecting to the same AI device at the same time are not supported at the moment.

3. How to build RTSP server?

Please prepare a server with Ubuntu OS:

Download easydarwin (https://github.com/EasyDarwin/EasyDarwin/releases), for example EasyDarwin-linux-8.1.0-1901141151.tar.gz

Run commands as follows:

```
tar -xzvf EasyDarwin-linux-8.1.0-1901141151.tar.gz
cd EasyDarwin-linux-8.1.0-1901141151
sudo ./easydarwin
```

Then you can using ffmpeg push stream to the url "rtsp://ubuntu-ip:554/test-001" then check url with vlc or potplayer, "test-001" is stream id, you can change it as you need.

This way, you can use a local video file to create an RTSP stream.

4. How to transcode a local video file to RTSP stream and how to get ffmpeg?

If you use Ubuntu, you can install ffmpeg by "sudo apt-get install ffmpeg", and if you use Windows, you can download ffmpeg in url "https://www.filehorse.com/download-ffmpeg-64/".

Below is demo command which is transcoding local file to RTSP stream:

Aupera Technologies Inc.

```
$ ffmpeg -threads 2 -re -fflags +genpts -stream_loop -1 -i  .\test.mp4 -c copy -f rtsp
rtsp://ubuntu-ip:554/test-001
```