

Horision Blurring

Horision Blurring requires Horus System V2 and Celantur to be installed on the same machine.

Horus System V2

With the Horus System V2 Batchcassing tool the user can process sets of recordings. The primary process methods of the Batch Pro Processing tool are described below. The Batch Processing tool allows the user to define if it is to anonymize entire vehicles and persons, or to just anonymize faces and license plates shown on the recordings. This process can be done to JPEG images or Ladybug PGR files.

Celantur

Celantur is the blurring engine used by Horus System V2 during the batch process.

In this document

1. System Requirements
2. Installation
 - 2.1. Install Celantur
 - 2.2. Install Horus System V2
3. Run a Blurring Batch Job
4. Celantur Deployment Instructions

1. System Requirements

- Ubuntu 20.04 system
- NVIDIA GPU: minimum 4GiB RAM, CUDA compute capability 3.7

- NVIDIA drivers installed (Ubuntu can do this during installation)
- Horus System V2 and blurring service (Celantur) must be deployed on the same system.
- Licenses will be required for
 - Celantur, the GPU ID is required to request a license to Celantur, see step 4.2 on how to find it.
 - Horus System V2, you can use your Horus license key to activate. Send the license request to support@horus.nu

2. Installation

2.1 Install Celantur

Please follow steps in section 4. to install Celantur on your system. For further information, please refer to the [Celantur on-premise installation](#) manual.

Please contact Celantur if additional support is needed.

2.2 Install Horus System V2

1. Copy the “Horus_Linking_Lab-*-x86_64.AppImage” to the system.
If you do not have this file, contact support@horus.nu
2. Mark the .AppImage as executable

This can be done via the *File Manager*.

Open the *File Properties* of the file and enable the *executable* setting.

Alternatively, this can also be done via the Command Line.

Open a command line and locate the .AppImage.

To give executable permission to the file enter the following:

```
chmod +x <Name of the AppImage>
```

Verify that the file now has executable permissions by entering:

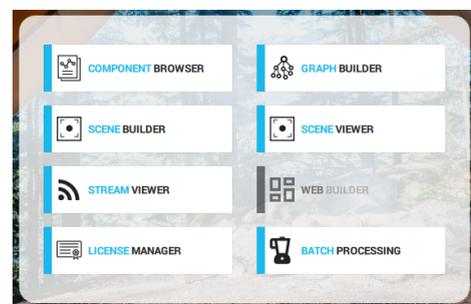
```
ls -l <Name of the AppImage>
```

The file should now have the same permissions as shown on the image below.

```
[13:56] lute @ ubuntu-gnu-linux: ~/Downloads
$ ls -l Horus_Linking_Lab-19.11.22-x86_64.AppImage
-rwxrwxr-x 1 lute lute 330661864 nov 26 13:45 Horus_Linking_Lab-19.11.22-x86_64.AppImage
[13:56] lute @ ubuntu-gnu-linux: ~/Downloads
$
```

3. Start the “Horus_Linking_Lab-*-x86_64.AppImage”
4. A window will open with the connection screen, select ‘This System’ to continue.
5. On the Main Menu open the License Manager. Enter the received License Key. Verify that the required components are now licensed via ‘List of License Components’.

The system is now ready for Batch Processing.



3. Run a Blurring Batch Job

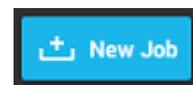
1. Verify that the Celantur blurring service is running.
See step 3 [chapter 4.3](#) on how to verify this.
2. Start the “Horus_Linking_Lab-*-x86_64.ApplImage”.
3. Connect to ‘This System’.



4. Open the ‘Batch Processing’ tool on the main menu.

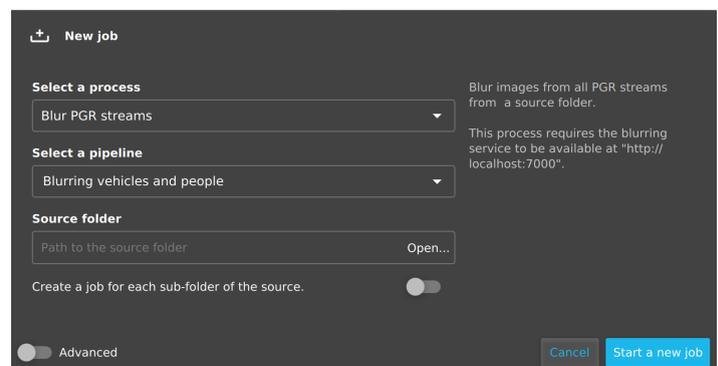


5. Select ‘New Job’ to start a batch job.
 - a. Select a process
 - Blur JPEG Images
 - Blur PGR Images



- b. Select a pipeline
 - Blurring vehicles and people
 - Blurring license plates and faces

This setting is only used for blurring.



New job

Select a process
Blur PGR streams

Select a pipeline
Blurring vehicles and people

Source folder
Path to the source folder

Blur images from all PGR streams from a source folder.
This process requires the blurring service to be available at "http://localhost:7000".

Create a job for each sub-folder of the source.

Advanced

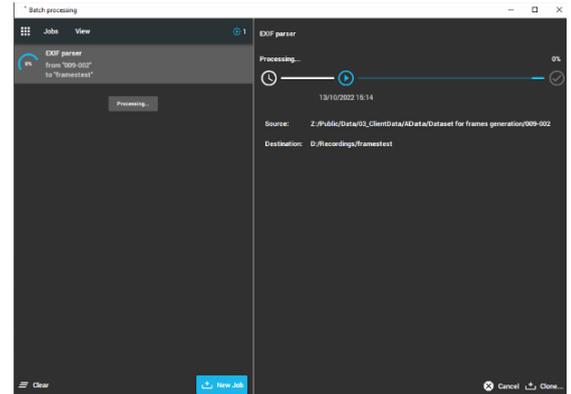
- c. Source folder
Select the path of the parent directory where the recordings are stored on the machine.
 - d. *create a job for each sub folder of the source*
Choose to create a separate batch job for each descendant sub-folder of the source location.
 - e. Destination folder
Select the path to the location on the machine where the new data will be written.
 - f. Start new job
Press on the blue ‘start new job’ button to add the batch job to the queue.

6. Monitor the status of the batch job.

If there are no pending jobs currently being processed or in the queue, then the new job will automatically start processing.

By selecting a batch job, you can view additional information about the processing.

When the processing is done the job will be marked as completed.



7. Check the output of the batch job.

Verify that the process has correctly processed the data by checking the destination folder.

Verify that vehicles and/or persons are now blurred.

4. Celantur Deployment Instructions

4.1 System Preparation

1. Install Docker. Following their instructions for [Ubuntu](#).
2. Install the Nvidia container toolkit with these [instructions](#).
3. Reboot the system before proceeding.

4.2 Celantur License

Contact Celantur to acquire a download password and runtime license for Celantur. (applicable only for blurring)

For this request a GPU UUID is required, this can be found by entering the following into the Command Line:

```
nvidia-smi -L
```

4.3 Deploying Celantur Docker Container

1. Acquire the Docker Container
In a Command Line enter the following:

```
mkdir -p ${HOME}/.config/horus/celantur && cd  
${HOME}/.config/horus/celantur
```

```
wget  
https://celantur-web.s3.eu-central-1.amazonaws.com/scripts/celantur.sh
```

```
export PASSWD=<passwd from celantur>
```

<passwd from celantur> Should be the password from Celantur.

```
mkdir licensing
```

```
cp <license file> licensing/
```

<License File> Enter the path to the license file for Celantur here.

```
bash celantur.sh --update
```

2. Create a Docker Compose file.

Go to `${HOME}/.config/horus/celantur` and create a `docker-compose.yml` file.

Give this file the following content:

```
services:  
  celantur:  
    image: celantur-on-premise:latest  
    ports:  
      - "7000:7000"  
    volumes:  
      -  
        ${HOME}/.config/horus/celantur/licensing:/home/celantur/licensing  
    restart: always  
    deploy:  
      resources:  
        reservations:  
          devices:  
            - capabilities: [gpu]  
    command: ["-l", "/home/celantur/licensing/license",  
              "--api", "0.0.0.0:7000",  
              "--api-endpoint", "http://localhost:7000/",  
              "-a", "face",  
              "-a", "license-plate",  
              "-a", "person",  
              "-a", "vehicle",  
              "--format", "whole"]
```

3. Start the new Celantur container.

```
cd ${HOME}/.config/horus/celantur
```

```
docker compose up -d celantur
```

Give the Container some time to start (~20s), you can verify if the container is running with the following command:

```
curl -X GET http://127.0.0.1:7000/v1/status
```

If the container is running a 'System Running' message will be printed to the Command Line.