Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults and associated risk the responsibility entirely of the Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer’s responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice’s product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.
Contents

Acronyms in This Document.................................................................................................................. 8
1. Introduction........................................................................................................................................... 9
  1.1. Pre-requisites ................................................................................................................................. 9
    1.1.1. Hardware Requirement.............................................................................................................. 9
    1.1.2. Software Requirements........................................................................................................... 9
  1.2. Limitations or Known Issues.......................................................................................................... 10
  1.3. Stop and Remove Older Version of Webapp ............................................................................. 10
2. sensAI Studio Installation Configuration ..................................................................................... 11
  2.1. Public Dataset Configuration .................................................................................................... 11
  2.2. Server IP Configuration .............................................................................................................. 11
  2.3. HTTPS Support Configuration .................................................................................................. 11
  2.4. NFS Configuration ...................................................................................................................... 13
  2.5. Kubernetes Configuration .......................................................................................................... 14
3. Deploying sensAI Studio ................................................................................................................ 15
  3.1. Deployment ................................................................................................................................. 15
4. Getting Started ............................................................................................................................... 16
  4.1. Logging in the Web Application ................................................................................................ 16
  4.2. Logging Out of the Web Application ......................................................................................... 18
  4.3. Changing the Password ............................................................................................................ 18
5. Dashboard ........................................................................................................................................ 20
6. Project .............................................................................................................................................. 21
  6.1. Viewing Project Details .............................................................................................................. 21
  6.2. Updating Project Details ............................................................................................................ 22
  6.3. Adding Project .......................................................................................................................... 22
    6.3.1. Model Zoo ............................................................................................................................. 23
    6.3.2. Local Model .......................................................................................................................... 24
    6.3.3. Other Repo ........................................................................................................................... 28
  6.4. Project Graph .............................................................................................................................. 28
  6.5. Download Project Zip ............................................................................................................... 29
7. Workspace ....................................................................................................................................... 30
  7.1. Adding Public Workspace ........................................................................................................ 30
  7.2. Adding Custom Workspace ....................................................................................................... 31
8. Dataset ........................................................................................................................................... 32
  8.1. Viewing Dataset Details ............................................................................................................ 32
  8.2. Adding Dataset .......................................................................................................................... 33
    8.2.1. Public .................................................................................................................................... 33
    8.2.2. URL ....................................................................................................................................... 34
    8.2.3. AWS ...................................................................................................................................... 34
    8.2.4. GCP ....................................................................................................................................... 35
    8.2.5. Uploading Dataset ................................................................................................................. 36
    8.2.6. Stream Dataset .................................................................................................................... 37
      8.2.6.1. Manual Capture ............................................................................................................... 37
      8.2.6.2. Continuous Capture ....................................................................................................... 37
      8.2.6.3. Video Stream .................................................................................................................. 38
  8.3. Editing Dataset ........................................................................................................................... 38
  8.4. Visualizing the Dataset ............................................................................................................. 40
    8.4.1. Changing Existing Class ....................................................................................................... 41
    8.4.2. Resizing Existing Bounding Box ......................................................................................... 42
    8.4.3. Drawing a New Bounding Box ............................................................................................ 43
    8.4.4. Deleting a New Bounding Box ............................................................................................ 43
  8.5. Auto-Annotation ........................................................................................................................ 43
  8.6. Converting Annotation ............................................................................................................. 45
# Lattice sensAI Studio User Guide

## Application Note

### Troubleshooting

<table>
<thead>
<tr>
<th>Issue</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not Able to See Camera for Stream Dataset</td>
<td>47</td>
</tr>
<tr>
<td>Model Zoo Projects or Workspaces Are Not Available</td>
<td>48</td>
</tr>
<tr>
<td>Not Able to Open Application Web Page</td>
<td>49</td>
</tr>
<tr>
<td>Dataset append and train/val/test split</td>
<td>50</td>
</tr>
<tr>
<td>Dataset Augmentation</td>
<td>51</td>
</tr>
<tr>
<td>Dataset Conversion Sample</td>
<td>52</td>
</tr>
<tr>
<td>Dataset Auto-Annotation Sample</td>
<td>53</td>
</tr>
<tr>
<td>Jobs</td>
<td>54</td>
</tr>
<tr>
<td>Adding Training Job</td>
<td>55</td>
</tr>
<tr>
<td>Adding Neural Network Compiler Job</td>
<td>56</td>
</tr>
<tr>
<td>Adding Download Job Artefacts</td>
<td>57</td>
</tr>
<tr>
<td>Download Complete Job</td>
<td>58</td>
</tr>
<tr>
<td>Modifying a Job</td>
<td>59</td>
</tr>
<tr>
<td>Scheduling a Job</td>
<td>59</td>
</tr>
<tr>
<td>Job Metrics</td>
<td>60</td>
</tr>
<tr>
<td>Adding Metrics</td>
<td>61</td>
</tr>
<tr>
<td>Comparing Metrics</td>
<td>62</td>
</tr>
<tr>
<td>Downloading Dataset</td>
<td>63</td>
</tr>
<tr>
<td>Pre-Processing the Dataset</td>
<td>64</td>
</tr>
<tr>
<td>Processing the Dataset</td>
<td>65</td>
</tr>
<tr>
<td>Upditing the User Details</td>
<td>66</td>
</tr>
<tr>
<td>Creating a Group</td>
<td>67</td>
</tr>
<tr>
<td>Adding and Removing Group Members</td>
<td>68</td>
</tr>
<tr>
<td>Deleting a Group</td>
<td>69</td>
</tr>
<tr>
<td>Configuring Memory Management</td>
<td>70</td>
</tr>
<tr>
<td>Memory Management</td>
<td>71</td>
</tr>
<tr>
<td>AWS NFS Setup Guide</td>
<td>72</td>
</tr>
<tr>
<td>AWS Kubernetes (EKS) Cluster Creation</td>
<td>74</td>
</tr>
<tr>
<td>Technical Support Assistance</td>
<td>75</td>
</tr>
<tr>
<td>Revision History</td>
<td>77</td>
</tr>
</tbody>
</table>
Figures

Figure 2.1. Public Dataset Configuration .................................................................11
Figure 2.2. Public Dataset Path Selection Reference ...............................................11
Figure 2.3. Server IP Configuration .....................................................................11
Figure 2.4. Https Configuration for API PATH ......................................................12
Figure 2.5. Https Configuration in ltf.config ......................................................12
Figure 2.6. Https Configuration in docker-compose.yaml ....................................12
Figure 2.7. Web Server Frontend Port Proxy Configuration for https .............12
Figure 2.8. Web Server Backend Port Proxy Configuration for https ............13
Figure 2.9. Web Server TensorBoard Port Proxy Configuration for https ......13
Figure 2.10. Web Server JupyterLab Port Proxy Configuration for https ......13
Figure 2.11. NFS Configurations ........................................................................14
Figure 2.12. Kubernetes Configurations ..............................................................14
Figure 3.1. Deploying sensAI Studio .................................................................15
Figure 4.1. Login Page .........................................................................................16
Figure 4.2. Password Change Window ...............................................................16
Figure 4.3. User Volume Configuration ..............................................................17
Figure 4.4. Web Tour ...........................................................................................17
Figure 4.5. Sign Out Window ...............................................................................18
Figure 4.6. Change Password Option ...................................................................18
Figure 4.7. Change Password Window ................................................................19
Figure 5.1. Dashboard Page ................................................................................20
Figure 6.1. Project Page .......................................................................................21
Figure 6.2. View Project ......................................................................................21
Figure 6.3. Update Project ..................................................................................22
Figure 6.4. Add New Project ...............................................................................22
Figure 6.5. Clone Project ....................................................................................23
Figure 6.6. Pull Project .......................................................................................23
Figure 6.7. Create Local Model Project ..............................................................24
Figure 6.8. Local Project Training Example ........................................................26
Figure 6.9. Local Project TensorBoard Example ...............................................27
Figure 6.10. Local Project Freeze Example ........................................................27
Figure 6.11. Download Model JSON ....................................................................28
Figure 6.12. Project Graph ..................................................................................29
Figure 6.13. Project Page .....................................................................................29
Figure 7.1. Workspace Page ................................................................................30
Figure 7.2. Add Public Workspace ......................................................................30
Figure 7.3. Add Custom Workspace ....................................................................31
Figure 8.1. Dataset Page ......................................................................................32
Figure 8.2. Dataset View .....................................................................................32
Figure 8.3. Dataset Type Selection ......................................................................33
Figure 8.4. Create Public Dataset ........................................................................33
Figure 8.5. Create Dataset from URL ....................................................................34
Figure 8.6. AWS Credential File ..........................................................................34
Figure 8.7. AWS Dataset Creation .......................................................................35
Figure 8.8. GCP Credential File ..........................................................................35
Figure 8.9. GCP Dataset Creation .......................................................................36
Figure 8.10. Upload Dataset ................................................................................36
Figure 8.11. Manual Capture ...............................................................................37
Figure 8.12. Continuous Capture .........................................................................38
Figure 8.13. Video Stream ..................................................................................38
Figure 8.14. Edit Dataset ....................................................................................39
Figure 8.15. Dataset Edit Options ........................................................................39
Figure 13.1. Memory Management Option.................................................................................69
Figure 13.2. Memory Management Global Configuration ..........................................................69
Figure 13.3. Memory Management User Configuration .............................................................70
Figure 13.4. Memory Management Global Configuration Update..............................................70
Figure 15.1. Kubernetes Service Account, Roles and Namespace .................................................72
Figure 15.2. Kubernetes Service Account, Roles and Namespace Creation ..............................73
Figure 15.3. API Token................................................................................................................73
Figure 16.1. Boot Error during Database Setup ...........................................................................74
Figure 16.2. Dataset Structure Tool Tip .......................................................................................74
Figure 16.3. Dataset Structure Example – Kitti .........................................................................75
Figure 16.4. Chrome Treat Origin as Secure ................................................................................76
## Acronyms in This Document

A list of acronyms used in this document.

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphics Processing Unit</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>JSON</td>
<td>JavaScript Object Notation</td>
</tr>
<tr>
<td>ML</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>NN</td>
<td>Neural Network</td>
</tr>
</tbody>
</table>
1. Introduction

The Lattice sensAI™ Studio is a user interface-based tool for training, validating, and compiling Machine Learning (ML) models optimized for Lattice FPGAs. The tool makes it easy to take advantage of transfer learning to deploy ML models. It allows you to select from a range of models pre-trained to cover popular use cases, bring your own data for additional training, validate the quality of training using TensorBoard, and compile for Lattice FPGAs.

The features of the Lattice sensAI Studio are:
- Model Zoo with variety of models based on multiple architecture
- Option to bring your own model and environment into framework
- Easy to use dataset pre-processing such as labeling, annotation conversion, augmentation
- Model training, freezing, TensorBoard
- Different types of transfer learning
- Storage management
- Informative dashboard for better visualization
- Support to compile model for various Lattice provided FPGAs
- JupyterLab interface for code editing
- User management and collaboration
- DIY installation and setup with docker container on CPU/GPU PC/Server/Cloud
- Support for Distributed Machine Learning training on cloud using Kubernetes

This document describes the installation/configuration, usage, and troubleshooting of the Lattice sensAI Studio.

1.1. Pre-requisites

The hardware, software, connection, and general requirements for this demonstration are provided in the following sections.

1.1.1. Hardware Requirement

The software requires the following hardware component:
- PC with Ubuntu 16.04/18.04/20.04 x64 distribution

1.1.2. Software Requirements

This software product requires the following software components to be installed:
- NVIDIA Container Toolkit - to use NVIDIA GPUs for model training
- Docker-ce (version 20.10.2 or above)
- Docker-compose (version 1.27.4 or above)
- Web browser: Google Chrome (preferred) or Firefox
- Pull preferred docker images (to get process faster after webapp is started)
  - Training docker image
    - docker pull ghcr.io/latticesemi/ml-training:tf-1.14.0
  - Custom Ubuntu image
    - docker pull ghcr.io/latticesemi/ubuntu
  - Jupyter image
    - docker pull ghcr.io/latticesemi/jupyter
  - NN compiler (sensAI) (pull versions which are to be used)
    - docker pull ghcr.io/latticesemi/sensai:v4.1
    - docker pull ghcr.io/latticesemi/sensai:v5.0
  - Create logs volume (run bellow command to create volume)
    - docker volume create logs
  - Pull latest sensAI Studio docker images

© 2022-2023 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal. All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.
• docker pull ghcr.io/latticesemi/web:<version>
• docker pull ghcr.io/latticesemi/app:<version>
  For example:
  • docker pull ghcr.io/latticesemi/web:v1.4
  • docker pull ghcr.io/latticesemi/app:v1.4

1.2. Limitations or Known Issues
• None

1.3. Stop and Remove Older Version of Webapp
• Run the command below from the directory where the webapp is started previously.
  • docker-compose --env-file ltf.config down
• Run the command below to remove extra containers and volumes (optional: Required if you want to do clean setup again).
  • docker container prune
  • docker volume prune
2. sensAI Studio Installation Configuration

2.1. Public Dataset Configuration

- Configure the local dataset path in ltf.config file to make it accessible as a public dataset.

```
# Local dataset path which will go public for all users
DATASET_PATH=/local/dataset/path
```

Figure 2.1. Public Dataset Configuration

- Update the absolute path of local dataset parent at both locations as mentioned in the image above.

**Note:** This is only required if you want to make any dataset public.

- Check the reference image below for path selection. As Figure 2.2 shows, there are multiple datasets available in dataset folder. You can add parent path where all datasets are present.

Figure 2.2. Public Dataset Path Selection Reference

2.2. Server IP Configuration

- Server IP configuration is required to access the webapp. It enables webapp access through public IP or internal IP.

- Update the deployment machine IP variable named “BACKEND_SERVER_ADDRESS” in ltf.config.

**Note:** Please note that localhost or 127.0.0.1 does not work as the server IP.

```
# Backend server address and port
# Example: http://192.168.1.2 or https://example.com
# Note: Do not use localhost or 127.0.0.1
BACKEND_SERVER_ADDRESS=http://192.168.0.192
BACKEND_SERVER_PORT=5000
# Frontend server port
SERVER_PORT=6001
```

Figure 2.3. Server IP Configuration

2.3. HTTPs Support Configuration

This configuration is only required if the user wants to deploy sensAI Studio with https URL.

To run sensAI Studio with https, below changes are required apart from the configurations mentioned above.
• Change the API path as below in ltf.config:

```
# Backend server address
# Examples:
# BACKEND_SERVER_ADDRESS=http://192.168.1.2
# BACKEND_SERVER_ADDRESS=https://example.com
# Note: Do not use localhost or 127.0.0.1
# Note: Do not add '/' at end
BACKEND_SERVER_ADDRESS=app.domain.com
```

Figure 2.4. Https Configuration for API PATH

• Enable HTTPS in ltf.config as below:

```
# Flag to enable https support
ENABLE_HTTPS=true
```

Figure 2.5. Https Configuration in ltf.config

• Change API_PATH in docker-compose.yaml as below in the services/web section:

```
web:
  container_name: ltf_frontend
  restart: always
  build:
    context: ./frontend
    dockerfile: Dockerfile.dev
  ports:
    - '${SERVER_PORT}:6000'
  depends_on:
    - app
  environment:
    - ENABLE_HTTPS=${ENABLE_HTTPS}
    - API_PATH=${BACKEND_SERVER_ADDRESS}/api
    - API_PORT=${BACKEND_SERVER_ADDRESS_PORT}
```

Figure 2.6. Https Configuration in docker-compose.yaml

Here “/api” is where a request is passed from frontend to backend. Same needs to be configured in the web server configuration (for example, in nginx.conf).

Apart from the configurations above, a web server also needs to be configured to redirect request from API path to API port and vice versa. Below are some example configurations for setting up a proxy.

• Proxy ‘/’ to frontend server port (SERVER_PORT). See the example below:

```
upstream client {
    server 127.0.0.1:6007;
}

location /
{
    proxy_pass http://client;
}
```

Figure 2.7. Web Server Frontend Port Proxy Configuration for https
• Proxy API path (for example, /api) to backend server port (BACKEND_SERVER_PORT)

```yaml
upstream api {
  server 127.0.0.1:9000;
}
location /api {
  rewrite /api/(.*) /$1 break;
  proxy_pass http://api;
}
```

Figure 2.8. Web Server Backend Port Proxy Configuration for https

• Proxy TensorBoard port to /tensorboard

```yaml
location /tensorboard {
  set $port 6008;
  if ($uri ~ "^/tensorboard/(\d+)/") {
    set $port $1;
  }
  proxy_pass http://127.0.0.1:$port;
}
```

Figure 2.9. Web Server TensorBoard Port Proxy Configuration for https

• Proxy JupyterLab code port to /code

```yaml
location /code {
  set $port 6008;
  rewrite ^/(.+)/?$<port>/d+$ /$1/$2/$3 break;
  rewrite ^/(.*)/?$<port>/d+$ /$1/$2 break;
  proxy_pass http://127.0.0.1:$port;
}
```

Figure 2.10. Web Server JupyterLab Port Proxy Configuration for https

### 2.4. NFS Configuration

sensAI Studio supports NFS as an option for storage. If this option is enabled, all user data is stored in the NFS drive instead of the local system. This requires changing the following variables in ltf.config.

- Set NFS_VOLUME=true.
- Add NFS_SERVER_IP.
- Add NFS_SERVER_PATH_USERDATA.

**Note:** NFS must be accessible from the deployment server.
2.5. Kubernetes Configuration

sensAI Studio supports running Machine Learning training jobs on Kubernetes cluster. To use Kubernetes, NFS is mandatory as the data needs to be stored in shared path which is accessible from the deployment server and Kubernetes cluster.

- Change the following variables in ltf.config:
  - Set ENABLE_KUBERNETES=true.
  - Provide the Kubernetes cluster name.
  - Provide Kubernetes API server and port details.
  - Provide the namespace which is to be used for creating training pods.
    - Note: Application should have the required permissions in the given namespace.
  - Provide the username and API token to access Kubernetes cluster.

Note: The Kubernetes master node must be accessible from the deployment server.

Refer to AWS Kubernetes (EKS) Cluster Creation for example of creating cluster on AWS cloud.
3. Deploying sensAI Studio

This chapter describes how to deploy sensAI Studio.

3.1. Deployment

- Deploy the webapp with the command below:

  # docker-compose --env-file ltf.config up -d

```
ml-training-framework: $ docker-compose --env-file ltf.config up -d
Creating network "ml-training-framework_default" with the default driver
Creating postgres ... done
Creating ltf_backend ... done
Creating ltf_frontend ... done
ml-training-framework: $
```

Figure 3.1. Deploying sensAI Studio

**Note:** `-d` is for running app as daemon. To debug and get logs, run command without `-d`.

- Go to http://<SERVER_IP>:<SERVER_PORT> in the browser to launch sensAI Studio. For HTTPS cases, use the web address specified as BACKEND_SERVER_ADDRESS to access application.
4. Getting Started

This section provides information on how to use the Lattice Machine Learning Training Framework.

4.1. Logging in the Web Application

To log into the web application:

1. Enter `http://<SERVER_IP>:6001` in the web browser and the login page is displayed. The default credentials are listed as follows:
   - Email – admin@admin.com
   - Password – 12345678

   ![Login Page](image)

   **Figure 4.1. Login Page**

   **Note:** For the fresh install, the initial database setup takes some time. If you notice a login error at first login, wait for some time and try again.

2. Type in the new password and select the “I agree terms and conditions” box when you are asked to change the password during the first login.

3. Click on Change.

   ![Password Change Window](image)

   **Figure 4.2. Password Change Window**
4. If the login is successful, the user is directed to the volume configuration page. Users can provide their own NFS details if they want to store data on NFS. If the user selects default, the user data is stored in volume configured by the system admin in `docker-compose.yaml`.

![User Volume Configuration](image)

**Figure 4.3. User Volume Configuration**

5. After the volume configuration, the user is redirected to the Dashboard page. For first time users, you can view the web tour before moving to any other window. This helps you go through all the major parts of the web application.

![Web Tour](image)

**Figure 4.4. Web Tour**
4.2. Logging Out of the Web Application

To log out of the web application:
1. Click on the user panel on the top-right corner of the web application.
2. Click on SIGN OUT to log out from the web application.

![Sign Out Window](image)

4.3. Changing the Password

To change your password:
1. Click on the user panel on the top-right corner of the web application.
2. Click on CHANGE PASSWORD.

![Change Password Option](image)
3. Provide the new password and click on **Change**.
5. Dashboard

The Dashboard module has different kinds of graphs available. These graphs allow you and the admin user to have a better visualization of the statistics. By default, you are redirected to the Dashboard page after logging into the web application. You can also go to Dashboard at any time by clicking on Dashboard from all the page links available in the main menu.

These graphs provide different information about projects, workspaces, datasets, jobs, memory usage, and so on.

Figure 5.1. Dashboard Page
6. Project

The Project page contains all the existing projects you created.

![Project Page]

Figure 6.1. Project Page

6.1. Viewing Project Details

To view the project details:

1. Go to the Project page.
2. Click on the icon to view the project details based on project type, as shown in Figure 6.2.

![Project Details]

Figure 6.2. View Project
6.2. Updating Project Details

To update a project:

1. Go to the Project page.
2. Click on the 🖋️ icon.
3. Modify the required details and click on Update.

6.3. Adding Project

To create a new project, click on Add Project. It shows the available options to add a project.
- Model Zoo – From the Lattice Model Zoo repository.
- Local Model – You can upload your own model to create project.
- Other Repo – Currently, this option is not supported.
6.3.1. Model Zoo
To create a project from Model Zoo:

1. Enter the **Project Name** and **Project Description**.
2. If a project is part of a group, select the group from the drop-down menu. If no group is selected, it is created in your private space or in a selected group space.
3. Select **Model Zoo** as project type.
4. Click on **Clone** on the project that should be created. You can also filter the list using the **Advance Search** option.

![Figure 6.5. Clone Project](image)

5. Once the project is created, it is reflected in the **Project page**.

### Pulling the Latest Code
You can pull the latest changes from the Model Zoo by clicking on the **v** icon from the **Project page**.

![Figure 6.6. Pull Project](image)
6.3.2. Local Model

To create a project from your local repository:

1. Enter the **Project Name** and **Project Description**.
2. If a project is part of a group, select the group from the drop-down menu. If no group is selected, it is created in your private space or in a selected group space.
3. Select **Local Model** as project type.
4. Enter the **Model Name** and select the **Target Framework**, **Framework Version**, and **Dataset Type** from the drop-down menu.
5. Open and select the JSON file. This is optional since the details from the JSON file are automatically populated if the JSON is uploaded. You can download the sample JSON for reference, if required.
6. Click on **Attach Files** to upload the **Project Zip**.

![Figure 6.7. Create Local Model Project](image)

7. To create new sections, click on **Add Section**. These sections are reflected in the section list of the job created, and can also be used for Training, TensorBoard, and Freezing. For each section, you can provide the command to run by adding the **Command** field.

   The command can also have other required/optional parameters. These parameters can be provided by adding parameters under a specified section.

   To add a new parameter, click on **Add Parameter**.

   **Note:** There are different fields of parameters:

   - **Visible name** – This is the name which is provided to user during Job creation.
   - **Argument** – The argument provided with section command (\(--<\text{argument}>\)).
   - **Default Value** – Default value shown for parameter during the job creation.
   - **Type** – Type of argument. The possible options are listed below:

     - file – Used for execution script. This file parameter can be a log file, the input files needed while running section command.
     - str – String Parameter
     - int – Integer Parameter
     - dir – Directory. When dir is selected as type, the framework automatically selects the path based on the Sub Type.
       - If Sub Type is dataset, the framework automatically identifies the path of the selected dataset.
• If Sub Type is artefacts, the framework automatically identifies the path of the job artefacts. This is used for model output directory.
• If Sub Type is checkpoint, the framework automatically identifies the path of the job artefacts. This is used for checkpoint path. The artefacts and checkpoint path can be the same. Checkpoint subtype is used for transferring learning where pre-trained checkpoints are copied to.
• transfer learning – You can provide the link to the pre-trained checkpoint zip for transfer learning. This shows the option to do transfer learning from pre-trained checkpoints while creating the job.
• checkboxes – This can be used for layer freezing. This option can only be used from JSON upload. Refer to the sample JSON format for details on how to use this. All option values are passed to the command separated by a comma.
• workspace – This option is used for sections other than training (for example, freezing). This provides you an option to select the workspace other than the training workspace.

Figure 6.8, Figure 6.9, and Figure 6.10 show the reference examples for the Training, TensorBoard and Freezing sections.

8. Once all settings are completed, click on Save to create the project.
Figure 6.8. Local Project Training Example
Download Project JSON
You can also download the JSON for the local project. To do this, view the project details as mentioned in Viewing Project Details, and then click on Download JSON.
6.4. Project Graph

Project graphs help user to compare runs of different jobs for the given project. Follow below steps to view graphs of project for different job runs:

1. Go to the Project page.
2. Click on the icon for the required project.
3. Select the Metrics parameter(s).
4. Click on Apply. This shows the project graph as below (Figure 6.12).
6.5. Download Project Zip

To download project files:

1. Go to the Project page.

2. Click on the icon to download project zip.

Note: For AutoKeras model, refer to Lattice sensAI Neural Network Compiler User Guide (FPGA-UG-02052).
7. Workspace

The Workspace page contains all the workspaces you created. Initially, there is a default workspace.

![Workspace Page]

Figure 7.1. Workspace Page

7.1. Adding Public Workspace

To add a public workspace:

1. Click on Add Workspace.
2. Select the Public Workspace option.
3. Enter the Workspace Name and Description. Click on Create Workspace on the workspace that is to be created.
4. If a workspace is part of a group, select the group from the drop-down menu. If no group is selected, it is created in your private space or in a selected group space.
5. Once the workspace is created, it is reflected in the Workspace page.

![Add Public Workspace]

Figure 7.2. Add Public Workspace
7.2. Adding Custom Workspace

To add a custom workspace:

1. Click on Add Workspace.
2. Select the Custom Workspace option.
3. Enter the Workspace Name and Description.
4. If a workspace is part of a group, select the group from the drop-down menu. If no group is selected, it is created in your private space or in a selected group space.
5. Enter the Framework, Framework Version, and Docker Image URL.
6. Click on Save to create the workspace.

![Figure 7.3. Add Custom Workspace](image-url)
8. Dataset

The Dataset page contains all the datasets you created.

![Dataset Page]

Figure 8.1. Dataset Page

8.1. Viewing Dataset Details

To view the dataset details:

1. Go to the Dataset page.
2. Click on the icon to view the dataset details, as shown in Figure 8.2.

![Dataset View]

Figure 8.2. Dataset View
8.2. Adding Dataset

To add a dataset, click on Add Dataset. It shows the different types of datasets available:

- Public – Create dataset from dataset directory, which is shared while starting the web application in the docker-compose file.
- Upload Dataset – Create dataset by uploading file.
- URL – Download dataset from using link to dataset.
- AWS/GCP – Not supported in current release.

![Figure 8.3. Dataset Type Selection](image)

8.2.1. Public

To add a Public dataset:

1. Select the Public option.
2. Provide the Dataset Name and Description.
3. Select Label Type and Dataset Path. If you do not select Other as label type, click on Validate to validate the dataset.
4. Click on Save.
5. Once the dataset is created, it is reflected in the Dataset page.

![Figure 8.4. Create Public Dataset](image)
8.2.2. URL

To add dataset using Dataset URL:

1. Select the URL option.
2. Provide the Dataset Name and Description.
3. Select Label Type and provide the Dataset URL. If you do not select Other as label type, click on Validate to validate the dataset.
4. Click on Save.
5. Once the dataset is created, it is reflected in the Dataset page.

![Figure 8.5. Create Dataset from URL](image)

8.2.3. AWS

To add dataset from AWS S3 bucket:

1. Select the AWS option.
2. Provide Dataset Name and Description.
3. Select Label Type.
4. Provide Bucket Name and Dataset Path (Dataset path inside the AWS bucket).
5. Click on Attach Files and upload the credential file in .txt format. The format for the same is given below.

![Figure 8.6. AWS Credential File](image)

6. Tick the Copy dataset to local option to download the dataset (It might take some time depending on dataset size and network bandwidth). This is optional and required if you want to copy dataset to the user volume (to save training time).
7. Click on Validate to validate the dataset.
8. Click on Save.
9. Once the dataset is created, it is reflected in the Dataset page.
8.2.4. GCP

To add the dataset from Google Cloud Platform (GCP):

1. Select the GCP option.
2. Provide Dataset Name and Description.
3. Select Label Type.
4. Provide Bucket Name and Dataset Path (Dataset path inside the GCP bucket).
5. Click on Attach Files and upload the service account credentials in .json format. The format of the same file is given below.

```
"type": "service_account",
"project_id": "sensai-studio",
"private_key_id": "872337392572m7e83d6u47y59w6wggw8jygdpf4z1s6g645178.pem",
"private_key": "-----BEGIN PRIVATE KEY-----
MIIEowIBAAKCAQEA2b2DjzgEYD3eGz7F4LA22RjzjCvo8IDnJ/mCZfNl9PG5gS
8JFFIp9q72xZu0Bm8E1G39e2mWzc/
-----END PRIVATE KEY-----
",
"client_email": "default-lens8072@lens8072.iam.gserviceaccount.com",
"client_id": "119549227799546647320984",
"auth_uri": "https://accounts.google.com/o/oauth2/auth",
"token_uri": "https://accounts.google.com/o/oauth2/token",
"auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
"client_cert_url": "https://www.googleapis.com/oauth2/v1/client_idDetails/930478732380-lens8072.iam.gserviceaccount.com"
```

6. Tick the Copy dataset option to download the dataset (It might take time depending on dataset size and network bandwidth). This is optional and required if you want to copy dataset to the user volume (to save training time).
7. Click on Validate to validate the dataset.
8. Click on Save.
9. Once the dataset is created, it is reflected in the Dataset page.
8.2.5. Uploading Dataset

To upload dataset from your local machine:

1. Select the Upload Dataset option.
2. Provide the Dataset Name and Description.
3. Select Label Type. If you do not select Other as the label type, click on Validate to validate the dataset.
4. Click on Attach Files to attach the dataset file and wait for the upload to complete.
5. Click on Save.
6. Once the dataset is created, it is reflected in the Dataset page.

![Dataset Creation](image)

**Figure 8.9. GCP Dataset Creation**

![Upload Dataset](image)

**Figure 8.10. Upload Dataset**
8.2.6. Stream Dataset

To upload dataset from your local machine:
1. Select the Stream Dataset option.
2. Provide the Dataset Name and Description.

8.2.6.1. Manual Capture
1. Click on Start Manual Capture.
2. Select the camera from the drop-down menu. If the camera is not displayed in the menu, refer to Not Able to See Camera for Stream Dataset for troubleshooting.
3. Capture the required number of images.
4. Click on Save to save dataset.

![Manual Capture](image)

**Figure 8.11. Manual Capture**

8.2.6.2. Continuous Capture
1. Provide values for Capture interval, Capture Duration and Capture Duration Interval.
2. Click on Start Continuous Capture.
3. Click on Save to save dataset.
8.2.6.3. Video Stream

1. Select Stream Type as Video Stream.
2. Provide values for Capture interval and Capture Duration Interval.
3. Provide Stream Url as a working video link.
4. Click on Save to save dataset.

8.3. Editing Dataset

User can edit datasets to:

- Change train/val/test splitting ratio.
- Add more images to existing Upload Dataset option.

This option is only available for KITTI and VOC datasets.

Follow the steps below to edit a dataset:

---

Figure 8.12. Continuous Capture

Figure 8.13. Video Stream
1. Go to the Dataset page.
2. Click on the Edit icon from the dataset tile.

![Figure 8.14. Edit Dataset](image)

3. Attach zip file if the dataset needs to be appended.
4. Change the train/test/val split ratio if required.
5. The attached zip needs to be validated.
6. Click on Save to save the dataset.

![Figure 8.15. Dataset Edit Options](image)
8.4. Visualizing the Dataset

Once a dataset is created, you can use this option to visualize the dataset. This option is available for KITTI and VOC datasets.

To visualize the dataset:

1. Go to the Dataset page.
2. Click on the View dataset icon, as shown in Figure 8.16.

![Figure 8.16. View Dataset](image)

3. Click on Visualization.

![Figure 8.17. Dataset Visualization Option](image)

The window displays the dataset images with bounding boxes drawn, based on labels from the dataset (Figure 8.18). Different classes are displayed with different colors.
You have a different option to save the modified annotation, change the class, draw new boxes, resize the existing box, or delete the existing box in the top panel (Figure 8.19).

You can also jump to a specific image by clicking on the Go to image option. Different classes are marked with different box colors, which can be viewed from bottom panel.

8.4.1. Changing Existing Class

To change the existing class:

1. Click on the bounding box of the class you want to change. The drop-down menu of existing labels appears on the top panel, as shown in Figure 8.21.
2. Select the new class from the drop-down menu on the top-right corner of the page. If the required label is not present, click on Create New to create a new label.
3. Click on Save.
8.4.2. Resizing Existing Bounding Box

To resize an existing bounding box:

1. Click on the icon on the top panel if it is not selected.
2. Hover the mouse on any corner of the bounding box. Once the corner is detected, drag and drop the corner to resize the box.
3. Click on Save.
8.4.3. Drawing a New Bounding Box

To draw a new bounding box:

1. Click on the icon on the top panel if it is not selected.
2. Draw a box on the image.
3. Change the label as mentioned in Changing Existing Class section.
4. Click on Save.

8.4.4. Deleting a New Bounding Box

To delete a new bounding box:

1. Click on the icon on the top panel of the screen.
2. Delete the box which needs to be deleted by clicking on the X symbol.
3. Click on Save on the top panel to save changes.

![Figure 8.23. Dataset Visualization Delete Box](image)

8.5. Auto-Annotation

You can create a dataset from plain images and generate auto-annotated dataset using pre-trained models. To do this, perform the steps below:

1. Create a compressed file (zip/tar) from the images, and create a dataset as mentioned in the GCP section:
   a. Select the GCP option.
   b. Provide Dataset Name and Description.
   c. Select Label Type.
   d. Provide Bucket Name and Dataset Path (Dataset path inside the GCP bucket).
   e. Click on Attach Files and upload the service account credentials in .json format. The format of the same file is given below.
2. Tick the **Copy dataset to local** option to download the dataset (It might take some time depending on the dataset size and network bandwidth). This is optional and required if you want to copy dataset to the user volume (to save training time).

3. Click on **Validate** to validate the dataset.

4. Click on **Save**.

5. Once the dataset is created, it is reflected in the **Dataset** page.

6. Click on **View Dataset** on the dataset tile.

7. Expand the **Dataset Auto Annotation** option.

8. Click on **Start**. It displays the available models for auto-annotation.
9. Select the required model from the list and click on **Start**. Once annotation starts, the progress can be seen, as shown in **Figure 8.28**.

![Dataset Auto-Annotation Model Selection](image)

**Figure 8.27. Dataset Auto-Annotation Model Selection**

You can refresh the status by clicking on the **Refresh** icon.

**Note:** The current model only works on plain images. The dataset type may change after annotation based on the model script. Therefore, re-annotation may not work once auto-annotation is done.

### 8.6. Converting Annotation

You can convert dataset from one type to another type. To convert an annotation:

1. Click on **View Dataset** on the dataset tile.
2. Expand the **Annotation Conversion** option.
3. Select the current label format as **From** and the required label format as **To**.

4. Click on **Start**.

5. Once conversion starts, the progress can be seen in the window, as shown in **Figure 8.31**.

You can refresh the status by clicking on the **Refresh** icon.
8.7. Downloading Dataset

To download the dataset, click on Download from the dataset tile, as shown in Figure 8.32.

![Figure 8.32. Dataset Download](image)

8.8. Pre-Processing the Dataset

There are different options available in the framework to process data like augmentation, conversion, and annotation. However, if you want to do some custom processing which is not available in framework, you can do it using the Dataset Pre-processing option.

For custom dataset pre-processing, you need to have a docker image with the required scripts and environment to do the processing. You can create a custom workspace; for the details, refer to the Adding Custom Workspace section.

Once the workspace is created, you can perform the steps below to pre-process the dataset:

1. Open the dataset, and go to the Dataset Preprocessing option.
2. Select the workspace, which has the required script and environment to run it.
3. Enter the command to run. Note that only a single command can be run at a time. You can run multiple commands by using shell options, like combining multiple commands with && or providing them with semicolon separated.
4. Click on Start.

![Figure 8.33. Dataset Pre-Processing](image)

You can click on the Refresh icon to refresh the processing status. You can also visualize logs by clicking on the Log button (if required).
Notes:
- User Dataset is available in `/lattice/dataset/<dataset-name>` location. For example, if dataset name is `human_dataset`, the root directory is `/lattice/dataset/human_dataset`.
- You can use your projects paths from this option using the same way as a dataset (for example, `/lattice/projects/<proj-name>`).

8.8.1. Dataset Conversion Sample
The dataset pre-processing can also be used for auto-annotation. You can bring your workspace with the required script, which can convert dataset from one format to another.

To do auto-annotation using your own model, perform the steps below:
1. Create the docker image, with the modifications below, and host it on the public docker repository:
   - You need to create a script to do the conversion from one format to another format. This script should be able to take the input path and output path as input so it can work on any dataset name used to create the dataset in the Lattice sensAI Studio.

   **Note:** You can skip this if a script is already present somewhere (for example, projects and any other workspace) and have all the packages to run the script.

2. Create the custom workspace in the Lattice sensAI studio from the custom docker image.
3. Create the dataset in the Lattice sensAI studio with input format by using the **Upload Zip** option.
4. Click on the **Dataset Pre-processing** option.
5. Select the workspace and provide the command to run for annotation.
6. Run the dataset pre-processing.

   **Note:** Dataset format maintained in framework is not changed during the dataset processing. For example, if the script converts the dataset from X format to Y format, the format specified during dataset creation does not change.

8.8.2. Dataset Auto-Annotation Sample
The dataset pre-processing can also be used for auto-annotation. You can bring your workspace with the required model and script which can label images.

To perform auto-annotation using your own model, perform the steps below:
1. Create docker image, with the modifications below, and host it on the public docker repository:
   - Frozen model for inference
   - You need to create a script to do inference on input images and create labels in required format. This script should be able to take input path and output path as input so it can work on any dataset name used to create dataset in Lattice sensAI Studio.

   **Note:** You can skip this if pb/script is already present (for example, in projects/jobs and any other workspace) and have all the packages to run the script.

2. Create the custom workspace in the Lattice sensAI studio from the custom docker image.
3. Create the dataset in the Lattice sensAI studio with plain images by using the **Upload Zip** option.
4. Click on the **Dataset Pre-processing** option.
5. Select the workspace and provide the command to run for annotation.
6. Run the dataset pre-processing.

   For example, the Human Count SSD MV2 model needs a dataset conversion from VOC to pickle format. This project contains script to convert it. You can do the conversion by selecting Default Workspace with command as `python3 /lattice/projects/mv2ssd/process_dataset.py --data-dir /lattice/dataset/voc/`, where `mv2ssd` is the project name and `voc` is the dataset name.

   **Note:** The dataset format maintained in the framework is not changed during the dataset processing. For example, if the script converts the dataset from plain images to Kitti format, the format specified during the dataset creation does not change.
8.9. Dataset Augmentation

You can perform augmentation on the dataset by using the Dataset Augmentation option. You have the flexibility to select the types of augmentation, percentage of images to augment, and output dimension.

Once the workspace is created, you can perform the steps below to pre-process the dataset:

1. Open the dataset.
2. Select the Dataset Augmentation option.
3. Click on Output Dimension and enable/disable the types of augmentation as required.
4. Provide/Update the default values of augmentation, such as percentage of images to augment and other parameters.
5. Click on Start to begin the augmentation.

You can click on the Refresh icon to refresh the processing status.

8.10. Dataset append and train/val/test split

Refer to the Editing Dataset section to append dataset to change train/val/test split.
8.11. Dataset Merging

The user can merge multiple datasets into one dataset.

**Note:** Only datasets with the same label type can be merged (Currently only KITTI/VOC supported).

To merge datasets:

1. Select the datasets to be merged.
2. Click on **Merge** on the top right corner, as shown in **Figure 8.35**.

3. Add **Dataset Name**.
4. Click on **Merge Dataset**, as shown in **Figure 8.36**.

**Figure 8.35. Merge Dataset in the Dataset Page**

**Figure 8.36. Merge Dataset Dialog**
9. Jobs

The Jobs Page contains all the jobs you created.

![Figure 9.1. The Job Page](image)

9.1. Adding Training Job

To add a training job:

1. Click on Add Jobs to create a new job.
2. Provide the Name.
3. Select the Project from the created projects, Workspace from the created workspaces, and Dataset from the created datasets. Based on the project, different job types are displayed in the Job Type drop-down menu.
4. Select Training for model training. Based on the selected project, the parameters are displayed with default values. You can change the parameter values, if required.
   For example, to change the checkpoint interval, change Checkpoint Interval parameter to required value.
5. Click on Save to start the job. To run the job later sometime, click on Save and Run.
Figure 9.2. Add Job

9.1.1. Logs

Once the job is started, the log appears in the log tab as shown in Figure 9.3. To stop the training (if required), click on Stop.
9.1.2. Freeze

To freeze the model:

1. Go to the Freeze tab in the Job window and it shows the freeze command.
2. Click on Freeze to freeze the model and run the freezing command, which displays the logs, as shown in Figure 9.5.

![Figure 9.3. Job Logs](image)

![Figure 9.4. Freeze Button](image)

![Figure 9.5. Freeze Logs](image)
9.1.3. TensorBoard

To open the TensorBoard, select the TensorBoard tab in the Job window and click on Launch. This displays the TensorBoard window, as shown in Figure 9.7.

![TensorBoard Launch Button](image)

**Figure 9.6. TensorBoard Launch Button**

*Note: If TensorBoard is not launched upon clicking on Launch, click on Refresh.*

![TensorBoard Window](image)

**Figure 9.7. TensorBoard Window**
9.2. Adding Neural Network Compiler Job

You can utilize the Lattice Neural Network Compiler features using the training framework web application. You can also analyze, compile, and simulate the trained model by choosing **NN Compiler** as the job type.

To add a neural network compiler job:

1. **Select NN Compiler as Job Type.**
2. Click on the appropriate command to run from the following options: **Analyze, Compile, and Simulate**. The Analyze option is a mandated step for compile and simulate.
3. Provide values for the other parameters required by NN Compiler such as **Framework, Class, Device, Model Files**, and so on. The user can also provide encrypted model file (elpb/elh5) as the input model file.

![Figure 9.8. Neural Network Compiler Job](image-url)
4. Click on **Save** to run the job. Based on the selected command, the application generates the firmware binary for the selected device and/or run hardware simulation. You can also see the logs in the log window, as shown in [Figure 9.10](#).

![Figure 9.9. Neural Network Compiler Options](image)

![Figure 9.10. Neural Network Compiler Logs](image)
9.3. Adding Download Job Artefacts

To download job artefacts, click on Download. This allows you to select the file to download. You can also select the appropriate file and save it on your chosen folder.

![Figure 9.11. Download Artefacts](image)

![Figure 9.12. Select Download File](image)

The user has the option to download the model as Encrypt Model (pb/h5) by selecting encryption and providing encryption password, as shown in Figure 9.13.
This encrypted model can be passed to Lattice sensAI compiler as it is to generate firmware binary. Lattice sensAI compiler can decrypt the model and parse it.

9.4. Download Complete Job

The user can download the complete job (dataset, project and artefacts).

To download the complete job, click on Download, select Full job checkbox, and click on Download. A zip file with the complete job containing a run.sh file is downloaded.

9.5. Modifying a Job

In case you make a mistake while creating a job, you have the option to edit the job instead of creating a new one. This helps avoid multiple unrequired jobs and save manual efforts in deleting these unrequired jobs.
9.6. Scheduling a Job

The user can schedule a job to be based on the specific pattern such as daily/weekly/yearly on a specific time and with a recurrence pattern of n days.

To schedule a job:
1. Click on Add Jobs to create a new job.
2. Provide the Name.
3. Select project from the created projects, workspace from the created workspaces, and dataset from the created datasets. Based on the project, different job types are displayed in the Job Type drop-down menu.
4. Select Training for model training. Based on the selected project, the parameters are displayed with default values. You can change the parameter values, if required.
5. For example, to change the checkpoint interval, change Checkpoint Interval parameter to required value.
6. Click on the Schedule.
7. Add details for scheduling such as Start Time, Recurrence Pattern, and the Range of recurrence for scheduling.
8. Click on Schedule.

![Schedule a Job](image)

Figure 9.15. Schedule a Job

9.7. Job Metrics

Metrics are useful to extract information such as loss, accuracy from training, and/or inference outputs. These metrics can be provided as regular expression by users. The Lattice sensAI Studio parses logs and store those metrics in the database. You can also compare metrics of different jobs using the job comparison feature.

9.7.1. Adding Metrics

Several Lattice-provided models have sample metrics added by default. You can add, modify, and/or remove depending on the requirement. You can also add metrics for your custom model while creating a model or running a job. Refer to the Local Model section to know more on how to add metrics during local project creation. For details on how to add metrics during job creation, refer to Adding Training Job.
9.8. Comparing Jobs

To compare jobs:

1. Go to the Job page.
2. Select the checkboxes for the jobs you want to compare. Maximum of three jobs can be selected for comparison.
3. Click on Compare. This opens the comparison window as shown in Figure 9.16. It compares job information such as Job Details, Parameters, and Metrics.

![Figure 9.16. Compare Jobs](image)

**Figure 9.16. Compare Jobs**

<table>
<thead>
<tr>
<th>Job Details</th>
<th>job_training_3</th>
<th>job_training_2</th>
<th>job_training</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>job_training_3</td>
<td>job_training_2</td>
<td>job_training</td>
</tr>
<tr>
<td>Date Time</td>
<td>May 20, 2023, 7:30 PM</td>
<td>May 20, 2023, 7:30 PM</td>
<td>May 20, 2023, 7:30 PM</td>
</tr>
<tr>
<td>Project Name</td>
<td>humanaccount_vgg</td>
<td>humanaccount_vgg</td>
<td>humanaccount_vgg</td>
</tr>
<tr>
<td>Job Type</td>
<td>training</td>
<td>training</td>
<td>training</td>
</tr>
<tr>
<td>Workspace Name</td>
<td>Default workspace</td>
<td>Default workspace</td>
<td>Default workspace</td>
</tr>
<tr>
<td>Dataset Name</td>
<td>dataset_humanaccount_224</td>
<td>dataset_humanaccount_224</td>
<td>dataset_humanaccount_224</td>
</tr>
</tbody>
</table>

![Figure 9.17. Job Details Comparison](image)

**Figure 9.17. Job Details Comparison**

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Dataset Type</th>
<th>Architecture</th>
<th>Dataset Path</th>
<th>Checkpoint Path</th>
<th>Image Set</th>
<th>Summary Steps</th>
<th>Main Steps</th>
<th>Checkpoint Interval</th>
<th>Visible Onuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>job_training_3</td>
<td>UTI</td>
<td>squeealoct</td>
<td>/lattic/database/dataset_humanaccount_224</td>
<td>/lattic/omelets/job_training_3</td>
<td>train</td>
<td>7</td>
<td>250000</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>job_training_2</td>
<td>UTI</td>
<td>squeealoct</td>
<td>/lattic/database/dataset_humanaccount_224</td>
<td>/lattic/omelets/job_training_2</td>
<td>train</td>
<td>7</td>
<td>250000</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>job_training</td>
<td>UTI</td>
<td>squeealoct</td>
<td>/lattic/database/dataset_humanaccount_224</td>
<td>/lattic/omelets/job_training</td>
<td>train</td>
<td>7</td>
<td>250000</td>
<td>7</td>
<td>0</td>
</tr>
</tbody>
</table>

![Figure 9.18. Job Parameter Comparison](image)

**Figure 9.18. Job Parameter Comparison**

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Step Losses</th>
<th>Cost Losses</th>
<th>Move Losses</th>
<th>Class Losses</th>
</tr>
</thead>
<tbody>
<tr>
<td>job_training_3</td>
<td>[ &quot;step_loss&quot;: &quot;2.0&quot; ]</td>
<td>[ &quot;cost_loss&quot;: &quot;0.1205000000000000000000&quot; ]</td>
<td>[ &quot;move_loss&quot;: &quot;0.7694106666666666666666&quot; ]</td>
<td>[ &quot;class_loss&quot;: &quot;0.0646850000000000000000&quot; ]</td>
</tr>
<tr>
<td>job_training_2</td>
<td>[ &quot;step_loss&quot;: &quot;2.0&quot; ]</td>
<td>[ &quot;cost_loss&quot;: &quot;0.1205000000000000000000&quot; ]</td>
<td>[ &quot;move_loss&quot;: &quot;0.7694106666666666666666&quot; ]</td>
<td>[ &quot;class_loss&quot;: &quot;0.0646850000000000000000&quot; ]</td>
</tr>
<tr>
<td>job_training</td>
<td>[ &quot;step_loss&quot;: &quot;2.0&quot; ]</td>
<td>[ &quot;cost_loss&quot;: &quot;0.1205000000000000000000&quot; ]</td>
<td>[ &quot;move_loss&quot;: &quot;0.7694106666666666666666&quot; ]</td>
<td>[ &quot;class_loss&quot;: &quot;0.0646850000000000000000&quot; ]</td>
</tr>
</tbody>
</table>

![Figure 9.19. Job Metrics Comparison](image)

**Figure 9.19. Job Metrics Comparison**
9.9. Job Graphs

Job graphs help the user to visualize the statistics of job run. The user can provide metrics to extract while creating a job and use those metrics to create graphs. Refer to the Job Metrics section for details on how to add metrics to job.

Once a job is complete, follow the steps below to create graphs:

1. Go to the Job page.
2. Click on the icon to open the graph window.
4. Click on Apply to generate the graph.

Figure 9.20. Job Graphs
10. Code

The **Code** page can be used to explore and update/modify the training code in projects. To navigate the **Code** page, the JupyterHub instance is launched and the interface is displayed as shown in **Figure 10.1**.

**Note:** If JupyterHub is not launched, click on **Refresh**.

![Figure 10.1. JupyterHub for Code](image-url)
11. User Management
The Admin User has the access to create other users and can assign different roles to users based on requirements. The possible roles are:

- **Admin** – All permissions and access.
- **Manager** – Permission to read and create things (projects, workspaces, datasets, and so on) in its own space or groups.
- **Viewer** – Read-only access (not supported).

**Note:** Creating more than one admin is possible, but is not fully-supported.

11.1. Creating a New User
To create a new user:

1. Click on the user panel on the top-right corner of the web application and select **USER MANAGEMENT**.

   ![User Creation Option](image1.png)

   **Figure 11.1. User Creation Option**

2. Click on **Add User**.

   ![Add User Option](image2.png)

   **Figure 11.2. Add User Option**
3. Provide the required details and click on **Save** to create the user.

![Figure 11.3. Create User](image)

**11.2. Removing a User**

To remove a user:

1. Click on the user panel on the top-right corner of the web application and select **USER MANAGEMENT**.
2. Click on the **Remove** icon of the user you want to delete.

![Figure 11.4. Remove User Option](image)
11.3. Updating the User Details

To update user details:

1. Click on the user panel on the top-right corner of the web application and select **USER MANAGEMENT**.
2. Click on the **Edit** icon of the user you want to modify the details, as shown in **Figure 11.5**.
3. Change the details needed and click on **Update**.

![Figure 11.5. Update User Option](image)

![Figure 11.6. Update User Details](image)
12. Group Management

The Group feature allows you to share projects, dataset, workspaces, and jobs with other users. The entities created in a group are stored in the group volume instead of your private volume. All members in a group can access them. When creating projects, dataset, workspaces, and jobs, you can choose where you want to create it. If no group is selected, it is created in your private space or in a selected group space.

*Note:* It is advised to use a shared project for shared jobs. Otherwise, all UI elements may not be visible properly to other users who do not have access to the project used in a shared job.

12.1. Creating a Group

To create a group:

1. Click on the user panel on the top-right corner of the web application and select **GROUP MANAGEMENT**. This displays all the groups available for the user.

![Figure 12.1. Group Management Option](image)

2. Click on **Add Group** to create the group.

![Figure 12.2. Group List](image)

3. Enter the group name and description. Select the users, depending on the requirement, from the drop-down menu.

4. Click on **Save** to create the group.
12.2. Adding and Removing Group Members

After creating a group, you can add or remove users when necessary.

To add or remove a user in a group:

1. Click on the user panel on the top-right corner of the web application and select GROUP MANAGEMENT.
2. Click on the Edit icon of the group you want to modify, as shown in Figure 12.4.

3. Select the users to add or remove, and click on Update as shown in Figure 12.5.
12.3. Deleting a Group

To delete a group:

1. Click on the user panel on the top-right corner of the web application and select **GROUP MANAGEMENT**.
2. Click on the **Delete** icon of the group you want to remove.

![Figure 12.6. Delete Group Option](image-url)
13. Memory Management

The Memory Management feature allows the periodic cleanup of past projects, datasets, and jobs. Based on configuration, older data gets deleted to free up space on the deployment server. This feature is configurable by the admin only and is very flexible. Data older than the configurable period is cleaned up automatically. This period can be configured globally or per user. It is also possible to keep selected items (project, dataset, and job) continually.

Note: This feature is currently in beta phase for this release.

13.1. Configuring Memory Management

To configure memory management:

1. Click on the user panel on the top-right corner of the web application and select SETTINGS. This opens the Memory Management Configuration window as shown in Figure 13.2.
2. To enable or disable the memory management feature, toggle **Global Setting**.

3. To enable or disable the memory management for project, dataset, and jobs respectively, toggle **Project**, and/or **Dataset**, and/or **Job** switches.

4. Select **Duration** period on how long you want to keep the items. Selecting **Forever** does not delete anything.

5. Check the **Run cleanup on save** box if you want to run the cleaner upon saving. Click on **Save**. By default, the cleanup scheduler runs once a day. Checking **Force settings for all users** remove user specific settings if already configured and applies global configuration for all users.

6. Click on **Save**. This runs the cleaner scheduler if **Run cleanup on save** is checked.

### 13.2. Memory Management User Setting

You can select a different period for individual users through the **User Setting** feature.

To set a different period:

1. Select the user from the drop-down menu. Click on **Add** to add user to the configuration list.

![Figure 13.3. Memory Management User Configuration](image)

2. Select duration and click on the icon to save the configuration. The icon switches to **Edit** as shown in **Figure 13.4**.

![Figure 13.4. Memory Management Global Configuration Update](image)

Once configuration is completed, the admin can remove the configuration by clicking on the **Delete** icon or modify it by clicking on the **Edit** icon.
14. AWS NFS Setup Guide

To setup NFS in AWS cloud, follow the links below:

1. Create a S3 bucket on AWS.
   - The user needs to create a S3 bucket to be used as storage for NFS (The user can also use volume/tape storage).

2. Create a S3 file Storage Gateway on AWS.
   - The user needs to create a relevant file gateway based on the above step for NFS creation.
   - Select m5.xlarge ec2 instance at least if using ec2 as compute option for Gateway. Also, allow access for ports 2049 and 22 while creating ec2 instance.

3. Create an NFS file share on AWS.
   - The user needs to create an NFS file share.
   - To access this NFS, the user needs to set the IP address of the ec2 instance and not file share.
15. AWS Kubernetes (EKS) Cluster Creation

- Create an Amazon EKS cluster.
- After the cluster creation, set up the service account and roles for the pod access from sensAI Studio. Connect to master node and run the following commands:
  - Create a file roles.yaml as given below. Change username, namespace, and other aspects as per requirement, same needs to be configured in Create an NFS file share on AWS.

![roles.yaml example]

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: my-namespace

apiVersion: v1
kind: ServiceAccount
metadata:
  name: my-user
  namespace: my-namespace

apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: my-role
  namespace: my-namespace
rules:
  - apiGroups:
      - 
        resources:
        - pods
        - pods/log
        - pods/status
        verbs:
        - get
        - list
        - create
        - update
        - delete

apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: my-binding
  namespace: my-namespace
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: my-role
subjects:
  - kind: ServiceAccount
    name: my-user
    namespace: my-namespace
```

Figure 15.1. Kubernetes Service Account, Roles and Namespace

- Run kubectl create -f roles.yaml.
  - This creates a role, namespace and user.
Figure 15.2. Kubernetes Service Account, Roles and Namespace Creation

- Get the api token by running these commands.
- SERVICEACCOUNT=<user-name>
- NAMESPACE=<name-space>
- APISERVER=$(kubectl config view --minify -o jsonpath='{.clusters[0].cluster.server}')
- SECRET_NAME=$(kubectl get serviceaccount -n $NAMESPACE $SERVICEACCOUNT -o jsonpath='{.secrets[0].name}')
- TOKEN=$(kubectl get secret -n $NAMESPACE $SECRET_NAME -o jsonpath='{.data.token}' | base64 --decode)
- curl $APISERVER/api/v1/namespaces/$NAMESPACE/pods --header "Authorization: Bearer $TOKEN" --insecure
- echo $TOKEN
- Copy the $TOKEN that is used for authentication.

Figure 15.3. API Token
16. Troubleshooting

16.1. Database Connection Failed during Boot

The user might see the messages below during boot for fresh setup. This is expected, as database initialization takes time. This error should go away after some time.

```
postgres | 2021-09-24 06:19:57.073 UTC [73] FATAL: database "saiml" does not exist
lfc_backend | error: database "saiml" does not exist
lfc_backend | at Parser.handlePacket (/usr/src/app/node_modules/pg-protocol/dist/parser.js:241:15)
lfc_backend | at Parser.parse (/usr/src/app/node_modules/pg-protocol/dist/parser.js:89:29)
lfc_backend | at Socket.<anonymous> (/usr/src/app/node_modules/pg-protocol/dist/index.js:41:38)
lfc_backend | at Socket.emit (events.js:314:20)
lfc_backend | at addChunk (_stream_readable.js:303:12)
lfc_backend | at readableAddChunk (_stream_readable.js:279:9)
lfc_backend | at Socket.Readable.push (_stream_readable.js:218:10)
lfc_backend | at TCP.onstreamRead (internal/stream_base_commons.js:188:23)
```

Figure 16.1. Boot Error during Database Setup

16.2. Dataset Validation Fails

Dataset validation fails if the uploaded dataset has a wrong folder structure. The user should upload the correct folder structure based on dataset type. The user can click on the Tool Tip icon (Figure 16.2) to get information on required structure.

Figure 16.2. Dataset Structure Tool Tip
Figure 16.3 shows a Kitti example.

Note: The user needs to make sure there is no extra folder when creating zip. In the above example, the user needs to create zip from ImageSets and the training folder (not from the KITTI_Dataset folder). So when extracted, it contains ImageSets and training in main dataset folder.

16.3. Not Able to Open Application Web Page
- Make sure IP address in ltf.config is correct.
  - It should be IP address of system in which application is deployed.
  - Make sure not to use localhost or local IP (for example, 127.x.x.x) as IP address.

16.4. Model Zoo Projects or Workspaces Are Not Available
Model Zoo projects and workspaces are listed based json maintained on GitHub. Make sure your system allows http/https file downloads.
You can run the command below to make sure it is accessible:
```
# wget https://raw.githubusercontent.com/LatticeSemi/Model_Info/dev/model_info.json
```
If you are unable to download it, contact your IT team to allow required access.

16.5. Not Able to See Camera for Stream Dataset
If sensAI studio is running in http mode, some browsers may not allow the access of microphone/camera. If the site is running on https, this issue should not be observed.
There might be workaround provided by browser to allow access of microphone/camera over non-secure connection as well. For example, to ignore Chrome secure origin policy, follow these steps.
• Navigate to `chrome://flags/#unsafely-treat-insecure-origin-as-secure` in Chrome.
• Find and enable the ‘Insecure origins treated as secure’ section (see below).
• Add any address you want to ignore the secure origin policy for. Remember to include the port number too (if required).
• Save and restart Chrome.

![Insecure origins treated as secure](image)

**Figure 16.4. Chrome Treat Origin as Secure**
Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/en/Support/AnswerDatabase.
## Revision History

### Revision 1.2, June 2023

<table>
<thead>
<tr>
<th>Section</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>Updated the description of the hardware component required by the software in the Hardware Requirement section.</td>
</tr>
</tbody>
</table>

### Revision 1.1, June 2022

<table>
<thead>
<tr>
<th>Section</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Updated for latest release.</td>
</tr>
</tbody>
</table>

### Revision 1.0, May 2021

<table>
<thead>
<tr>
<th>Section</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>