

Project Planning Worksheet

To pass this course, you'll need to create a project that matches this criteria:

"Based on your understanding of the material, you're required to build and submit an open-source project that uses NVIDIA Jetson and incorporates elements of AI (machine learning or deep learning) with GPU acceleration, along with a video demonstrating the project in action. For example, you could collect your own dataset and train a new DNN model for a specific application, add a new autonomous mode to JetBot, or create a smart home / IoT device using AI - these need not be limited only to topics covered in the course. For inspiration, see the [Jetson Community Projects](#) page - the possibilities are endless!"

To pass the certification, your project will be reviewed based on the following criteria:

- **AI** - The project uses deep learning, machine learning, and/or computer vision in a meaningful way and demonstrates a fundamental understanding of creating applications with AI. Factors include the effectiveness, technical complexity, and performance of your AI solution on Jetson.
- **Impact / Originality** - The concept of your project is novel and applies AI to solve or address challenges or issues faced by yourself or society. Also, our ideas and work are either original or derivative in a significant way.
- **Reproducibility** - Any plans, code, and resources needed for someone else to build and use the project are included in the repository and are easy to follow.
- **Presentation and Documentation** - The video effectively demonstrates and explains various aspects of the project, and there exists a clear, complete README in the repository that documents any steps needed to build/run the project, along with diagrams and images.

Follow these steps to plan out your project

Part One: Brainstorming

Write down 3-5 ideas for problems that you see in the world around you that you could create an AI to help solve. You can use [student example projects](#) or [community example projects](#) for inspiration or look back on past lessons that you enjoyed.

(MAYBE POSSIBLE) AI Posture Rating

- It uses Topology Mapping to rate an individual posture effectively
- We would need:
 - Topology of a good posture
 - Ability to detect how off landmarks in the provided photo are from topology on the good postures
 - Or use the confidence to tell how close the AI thinks the posture is from good/bad

(POSSIBLE) AI Age eligibility

<https://www.kaggle.com/datasets/mariafrenti/age-prediction/data>

- Use Facial recognition to try and sort an individual into a certain age range based on their face to maybe alleviate ID fraud or just quickly check using AI whether someone is being truthful of their age (within reason)
 - We would need a dataset for EACH age range that we use
 - A lot of training to avoid false sorts (within reason)
 - Consistency

(POSSIBLE) Traffic Sign Recognition

- Use Kaggle's Traffic Sign dataset
(<https://www.kaggle.com/datasets/tuanai/traffic-signs-dataset>)
- With a given picture of a traffic sign, return what it is: would make people more knowledgeable about the road signs which (by a small margin) will improve road safety and traffic flow
 - Kaggle Dataset
 - Make sure to train it a lot (returning the wrong sign would be no good)

Part Two: Details

Write down the answers to these questions for your **two favorite** ideas:

AI: How would the AI work? Technically speaking what kind of network is it and how does this network work?

Idea 1: Improve Poor posture by comparing full body photo topographies to a dataset of good posture topographies and bad posture ones (Use confidence of the AI's prediction to somewhat "Grade" their posture)

Idea 2: Use facial recognition to prevent people from getting away from lying about their age by detecting their age from the elaborate Kaggle dataset

Impact: What impact would this project have? Who does it impact and in what ways?

Idea 1: Improve posture, specifically in individuals who sit down often with poor desk and setting arrangements, or improve quality of life in general for people who don't practice correct posture

Idea 2: 4000 minors (under 21) die due to alcohol consumption, and 108,000 deaths a year from Overdoses, may be reduced slightly from fewer kids getting away with lying about their age and getting alcohol and drugs they shouldn't have

Part Three: Resources

Now that you have thought out the impact and technical aspects of how the AI will work, it is time to map out what resources are going to be needed to complete your project.

Docs from jetson-inference: Add your documentation or tutorial link below

Image Classification

- [Using the ImageNet Program on Jetson](#)
- [Coding Your Own Image Recognition Program \(Python\)](#)

Example code: Add your example code below

Jupiter Labs code is very similar to what I'm trying to achieve

```
import torch-vision.transforms as transforms
from dataset import ImageClassificationDataset
```

```
TASK = 'age'
```

```
CATEGORIES =
```

```
['10','11','12','13','14','15','16','17','18','19','20','21','22','23','24','25','26','27','28','29','30','31','32','33','34','35','36','37','38','39','40','41','42','43','44','45','46','47','48','49','50','51','52','53','54','55','56','57','58','59','60','61','62','63','64','65','66','67','68','69','70','71','72','73','74','75','76','77','78','79','80']
```

```
DATASETS = ['A', 'B']
```

```
# DATASETS = ['A', 'B', 'C']
```

```
TRANSFORMS = transforms.Compose([
    transforms.ColorJitter(0.2, 0.2, 0.2, 0.2),
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
    transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
```

```
)
```

```
datasets = {}
```

```
for name in DATASETS:
```

```
    datasets[name] = ImageClassificationDataset('../data/classification/' + TASK + '_' +  
    name, CATEGORIES, TRANSFORMS)
```

```
print("{} task with {} categories defined".format(TASK, CATEGORIES))
```

^ This code is able to recognize Singular aspects in photos (example:thumbs up vs thumbs down)

Train it using the bellow dataset from Kaggle

Make sure to Train it ALOT, the goal of this program is to be a simple way to try and prevent people from lying about their age, it would be detrimental if it gave wrong outputs and assisted that.

Datasets: If applicable, add the dataset that you will be using below

<https://www.kaggle.com/datasets/mariafrenti/age-prediction/data>

Miscellaneous: Add any other resources you might need for your project below.

Maybe if possible make the program easy to use and applicable without vs code terminal and all that

Part Four: Documentation

Video: Write down any key points that you want to add into your video below

- Talk about the deaths per year due to alcohol
- Talk about how the code works (simplistically so don't explain the individual functions like the pytorch and all that)
- Tell them the results
- Show a simple and funny example of the code functioning in its intended environment

Documentation: Write down any key points that you want to make sure are in your readme doc.

- The description of the code (Same thing as earlier but more detailed)
- Description of Jetson nano
- Tutorial on how to use
- Name and inspiration

Reproducibility: How could your project be reproduced or run on another machine? Make sure to remember all the steps that make your project work.

1. By replicating the Jupiter labs function theoretically it can be used on any device with a camera
2. The actual use case of the Program is very simplistic, just show the individual face on the camera and it will output the predicted age

3.

4.