

# 3D Human Pose Estimation

Presented by Yuan Cheng

# People

- Yuan Cheng – [chengyu@mail.uc.edu](mailto:chengyu@mail.uc.edu)
- Advised by Dr.Chia Yung Han – [han@ucmail.uc.edu](mailto:han@ucmail.uc.edu)

# Preview

- Project Description
  - Abstract
  - Constraints
  - User Stories
  - Design Diagrams
- Project Review
  - Current state of project
- Project Plan
  - End of term accomplishments
  - Milestones
  - Timeline
- Expo Demo
  - Demo

# Description - Project Abstract

- **Goal:**

Implementing a deep learning based human pose estimation application that predicts human pose from videos or camera stream in real-time.

- **Project Components:**

- Deep learning model training;
- Video processing application framework design.

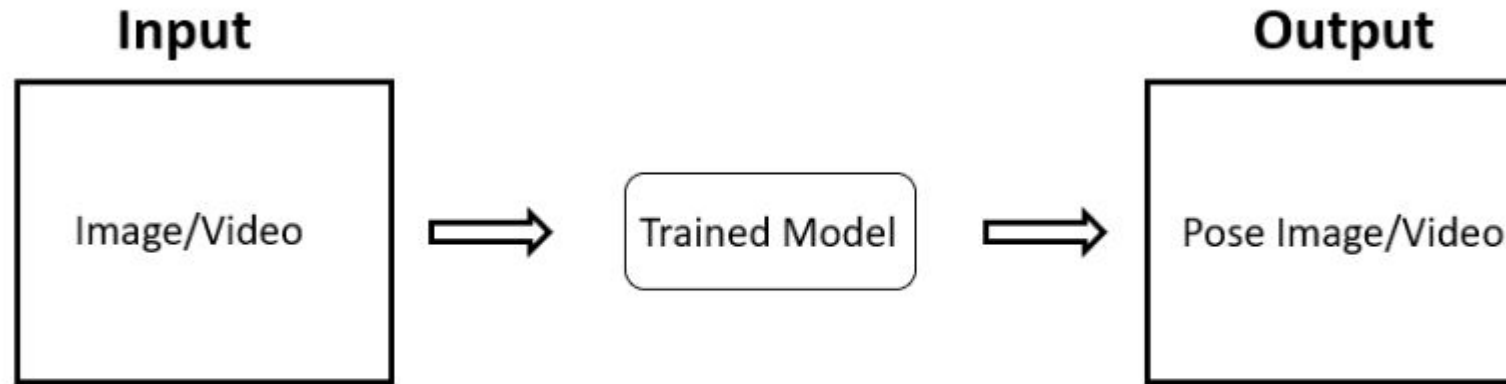
- **Keywords:**

Python, Deep learning, CNN, Pose estimation, Image processing, GUI.

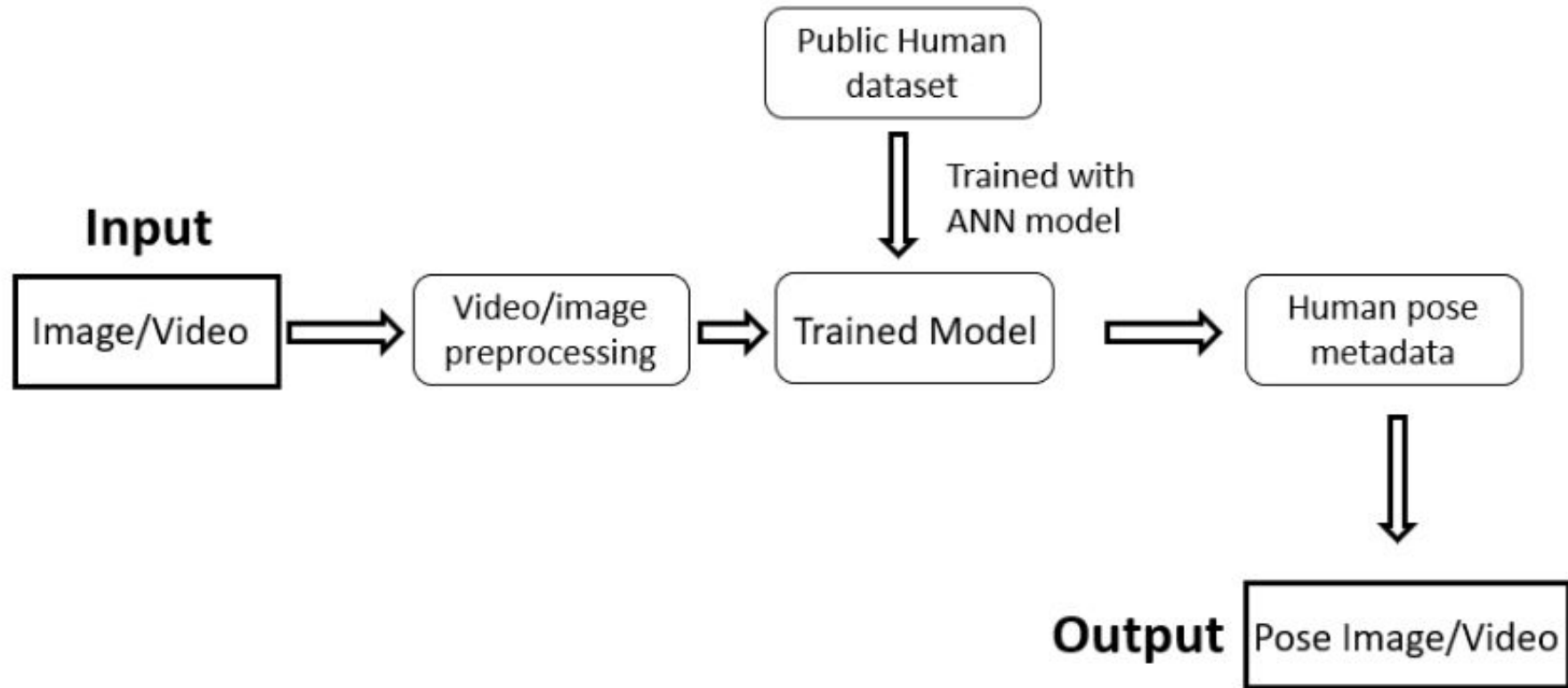
# Description - User Stories

- As a researcher, I want to utilize the metadata of human pose results from this project to analyze the relation between pose and activities.
- As an Animator, I want to extract human pose from real human activities so that I have an easier movement designing in animation.
- As a game maker, I want to have an accurate human skeleton extraction from videos so that I can ensure the accuracy of somatosensory sensor data in somatosensory games.

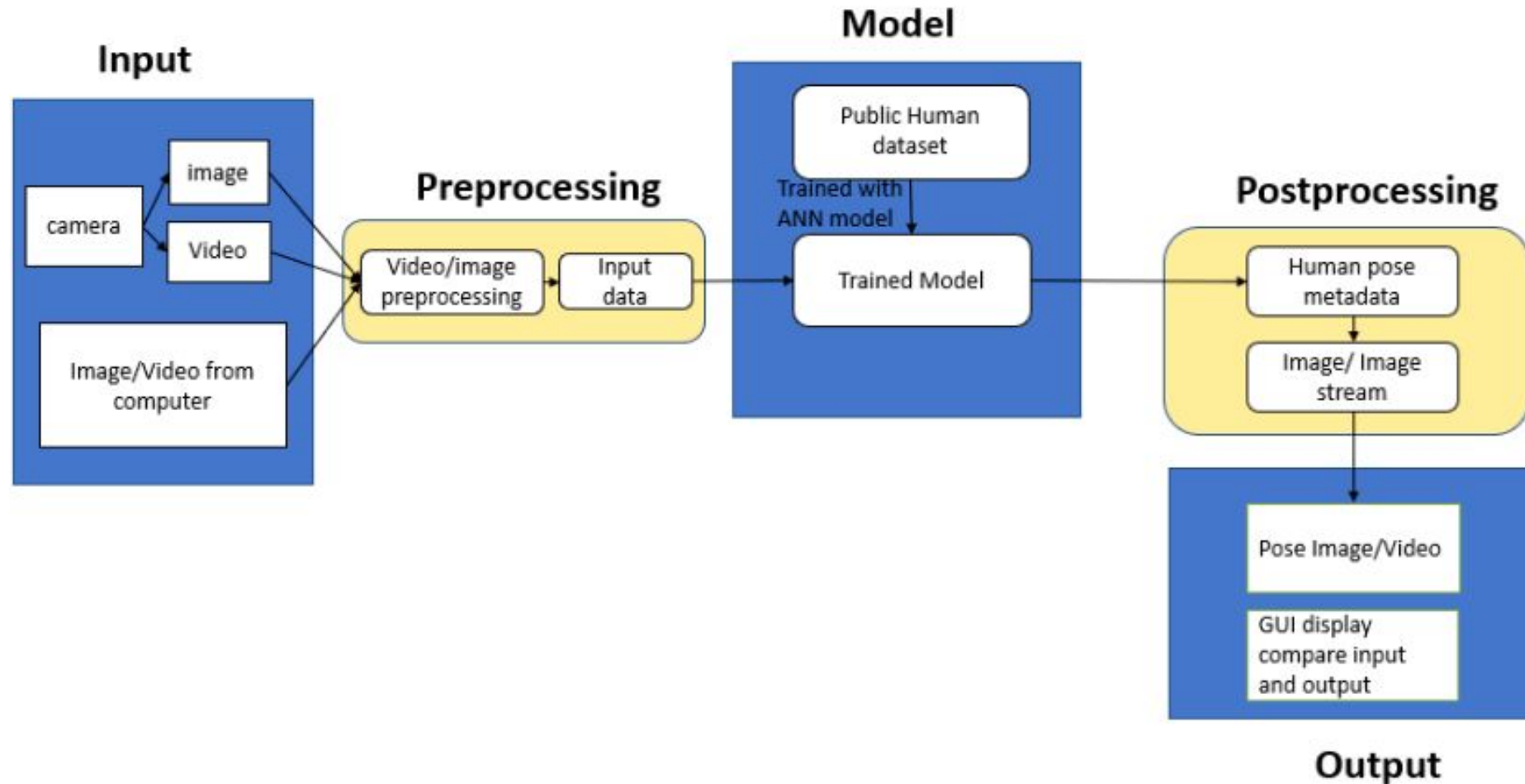
# Description – High Level Design Diagram



# Description – Detailed Design Diagram



# Description – More Detailed Design Diagram





# Description - Major Project Constraints

- Time - It is a great challenge that I have 5 other CS courses along with CS senior design this semester and I have to study and research materials for senior design project.
- Ethical and Legal - Video data needs to be carefully interacted with to avoid any privacy violation.
- Economic Costs - Requiring Video Recording equipment.

# Review - Project Progress: current state

## **Learn:**

- python framework;
- OpenCV image and video processing

# Plan - End of Term Accomplishments

- Finish learning the python framework and image processing.
- Research on the methods of preprocessing images/videos.
- Develop application framework.
- Write a preprocessing code.

# Plan – Project Milestones

1. Learn python framework and OpenCV image and video processing
1. Literature review of neural networks for human pose
1. Coding and Testing
1. Deployment and Documentation

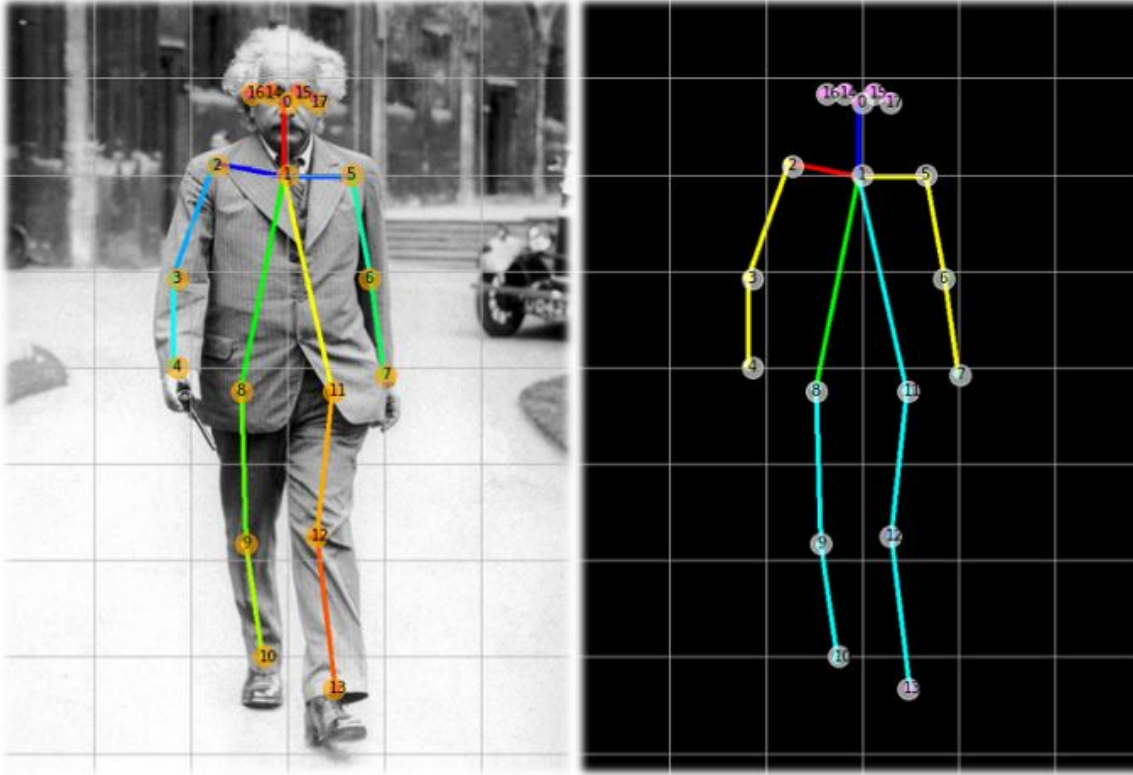
# Plan – Project Timeline: Fall 2020

#	Task	Yuan's Effort	Start Date	End Date	Milestone Category
1	Develop an application framework using python for the project.	100%	11.1.2020	11.7.2020	1
2	Learn image processing from video & camera using OpenCV and python.	100%	11.8.2020	11.14.2020	1
3	Research image preparation methods for pose estimation.	100%	11.15.2020	11.21.2020	1
4	Write a program for image & video acquisition and preprocessing.	100%	11.29.2020	12.12.2020	2

# Plan – Project Timeline: Spring 2021

#	Task	Yuan's Effort	Start Date	End Date	Milestone Category
5	Learn to write and train a neural network model from PyTorch.	100%	1/11/2021	1/17/2021	2
6	Research different neural networks models for human pose estimation	100%	1/18/2021	1/24/2021	2
7	Build a neural networks model using PyTorch	100%	1/25/2021	2/7/2021	3
8	Prepare public human pose datasets for training.	100%	2/8/2021	2/14/2021	3
9	Train the model with public human pose datasets.	100%	2/15/2021	2/28/2021	3
10	Compare the results from the trained model and fine-tuning the model.	100%	3/1/2021	3/7/2021	3
11	Test the model performance with some real scenes.	100%	3/8/2021	3/14/2021	3
12	Develop a GUI framework for selecting input sources and displaying results.	100%	3/15/2021	3/24/2021	4
13	Deploy the project system into a computer or an edge device.	100%	3/25/2021	4/1/2021	4
14	Document a user's manual of the project.	100%	4/2/2021	4/11/2021	4

# Expected Demo at Expo



The final demo expected to display the original video and estimated skeletons on the screen at the same time.

Image source: [PhysicsWorld — Einstein in Oxford \(1933\)](#)

Thank you for Watching!