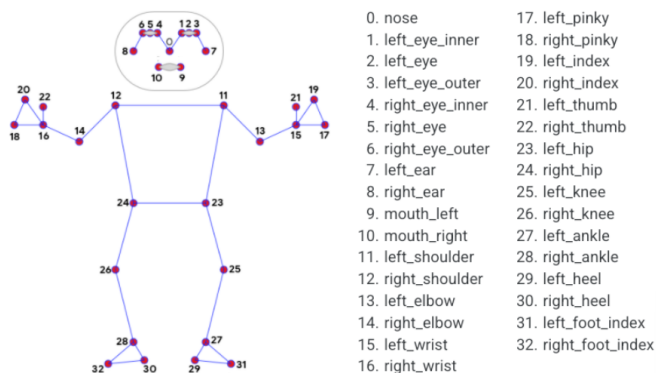


# Controlling Drone with Body Postures

- Human pose estimation from video plays a critical role in various applications such as sign language recognition, and full-body body pose control.

In this assignment, we try to control a drone using our body pose.

- MediaPipe Pose : two-step detector-tracker ML pipeline to detect 33 3D landmarks.



## 1) Let's start with implementing media pipes body key point detection model using webcam or static image.

<https://google.github.io/mediapipe/solutions/pose.html>

For each frame processed we have an array of 3d coordinates of 33 key points. We use these key-points for body pose recognition

You can use the pretrained model for body pose recognition or create your own dataset.

- The dataset can be created by running body key point detection for certain time and save different body poses.

## 2) You can also first save your body pose as video and convert video into frames using the [videos2frame.py](#) script

It is recommended we save the dataset in the below format for training.

## Datasets

## Train

## Gesture1

frame1.png

frame2.png



..

## Gesture-n

frame1.png

frame2.png



• •

3) For each frame we run the body key point detection and append it to the CSV file. You can refer [my\\_pose\\_train.py](#) script to finish the task.

At the end of step3 you should have a CSV with information about key points and its label.

[illegible]

**4) Now we are ready with our dataset, we now build a KNN classifier to recognize our body poses.**

Train X = N \* [(x0,y0), (x1,y1) ..... (x32,y32)]

Train  $Y = N \times \text{label for each body pose}$

You can refer to [knncvs.py](#) to train the KNN classifier and save the model as a pickle file.

By the end of step 4 we should have a pickle file saved in your directory.

**5) Let's test the classifier on real time. You can refer to [PoseModule.py](#) to test our classifier out.**

You should be able to perform body key point detection and body pose recognition in real-time.

**6) Now all is left to implement drone control using our body poses. You can refer to [drone\\_pose\\_control.py](#)**

You can take this further to make the drone do some cool actions or trajectories with your pose.