

# REP COUNT



# CHALLENGE?



***No simple or affordable way to automatically count reps from regular videos.***

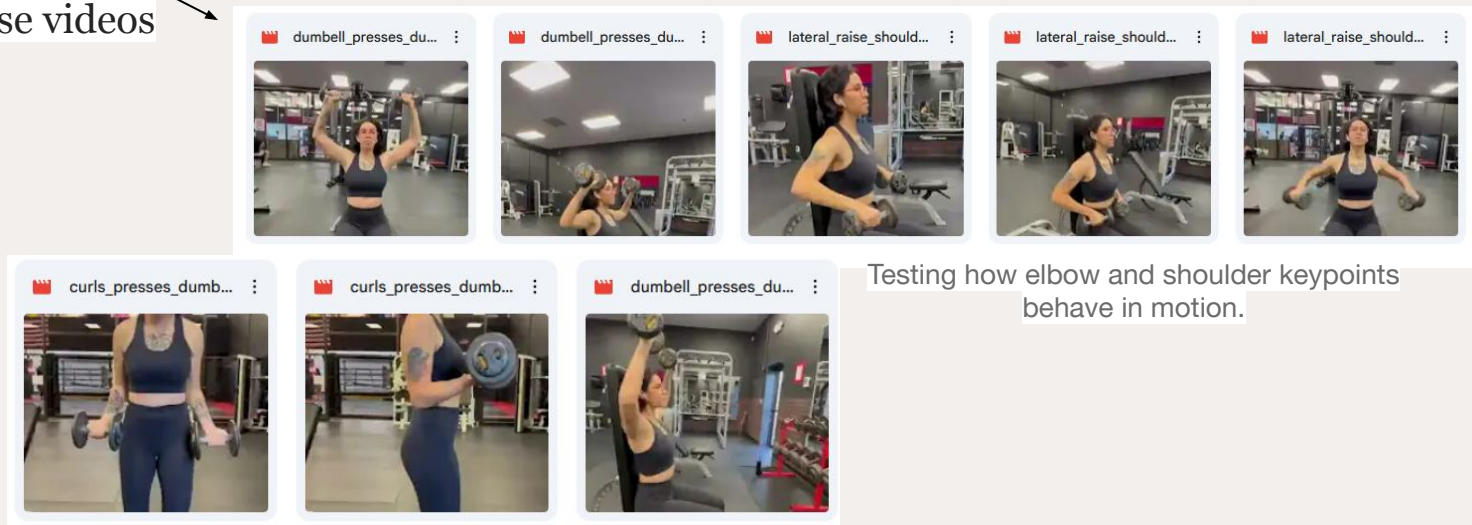
***People/trainers/coaches rely on manual counting, human error, slow feedback.***

***Challenging to track form or progress when training alone.***

# 1 – CAPTURING

## CAPTURING - COLLECTING

I recorded a  
small batches of  
exercise videos

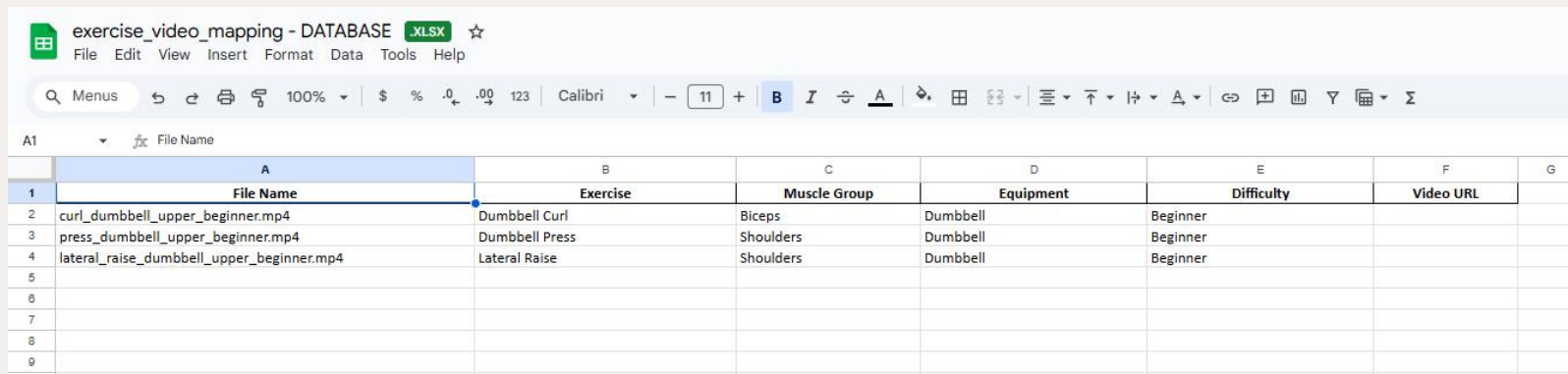


Testing how elbow and shoulder keypoints  
behave in motion.

I recorded each exercise from multiple angles because the model struggles to consistently recognize limbs and body extremities mediapipe still has limitations, especially when the movement is fast, the angle is unusual, or the lighting isn't ideal.

# 2 – mini DATABASE

## database



The screenshot shows an Excel spreadsheet titled "exercise\_video\_mapping - DATABASE" with a green .xlsx icon and a star. The ribbon includes File, Edit, View, Insert, Format, Data, Tools, and Help. The status bar at the bottom indicates "A1" and "File Name". The spreadsheet contains a table with the following data:

	A	B	C	D	E	F	G
	File Name	Exercise	Muscle Group	Equipment	Difficulty	Video URL	
1							
2	curl_dumbbell_upper_beginner.mp4	Dumbbell Curl	Biceps	Dumbbell	Beginner		
3	press_dumbbell_upper_beginner.mp4	Dumbbell Press	Shoulders	Dumbbell	Beginner		
4	lateral_raise_dumbbell_upper_beginner.mp4	Lateral Raise	Shoulders	Dumbbell	Beginner		
5							
6							
7							
8							
9							

each file with its exercise name, equipment, muscle group, difficulty, and a video URL. This helps me stay structured as I start my first tests with pose estimation.

# 3 – SCRIPTS



---

## scripts

### `view_landmarks_video.py` – *Exploration*

First I validated that MediaPipe Pose was working and that I could see the skeleton.

### `extract_keypoints.py` – *Data/debug* (nice but optional)

Then I created a script to export landmarks/angles to CSV so I could inspect the numbers and later use them for ML or debugging.

### `rep_counter.py` – *Script*

Finally I used those landmarks + angle thresholds to implement a rep counter in real time.

## scripts

lateral\_raise\_shoulder\_dumbbells\_beginner\_front\_view.mp4

- curls\_presses\_dumbbells\_beginner\_front\_view.mp4
- curls\_presses\_dumbbells\_beginner\_lateral\_view.mp4
- dumbbell\_presses\_dumbbells\_beginner\_front\_view.mp4
- dumbbell\_presses\_dumbbells\_beginner\_lateral\_view.mp4
- dumbbell\_presses\_dumbbells\_beginner\_semi\_lateral\_view.mp4
- lateral\_raise\_shoulder\_dumbbells\_beginner\_front\_view.mp4**
- lateral\_raise\_shoulder\_dumbbells\_beginner\_lateral\_view.mp4
- lateral\_raise\_shoulder\_dumbbells\_beginner\_semi\_lateral\_view.mp4

lateral\_raise\_landmarks\_output.mp4  
lateral\_raise\_shoulder\_dumbbells\_beginner\_lateral\_view\_keypoints.csv  
extract\_keypoints.py  
README.md  
rep\_counter.py  
view\_landmarks\_video.py

OUTLINE

- VIDEO\_FILENAME
- SCRIPT\_DIR
- VIDEOS\_DIR
- VIDEO\_PATH
- mp\_drawing

TIMELINE rep\_counter.py

- File Saved now
- File Saved 1 min
- File Saved 24 mins

extract\_keypoints.py view\_landmarks\_video.py rep\_counter.py X

```
1 import cv2
2 import mediapipe as mp
3 import numpy as np
4 import
5
6
7 VIDEO_
8
9
10 SCRIPT
11 VIDEOS
12 VIDEO_
13
14 print(
15 print(
16
17 mp_dre
18 mp_pos
19
20
21
22 def ca
23 ...
24 a
25 b
26 c
27
28 ra
29
30 an
31 if
32
33 re
```

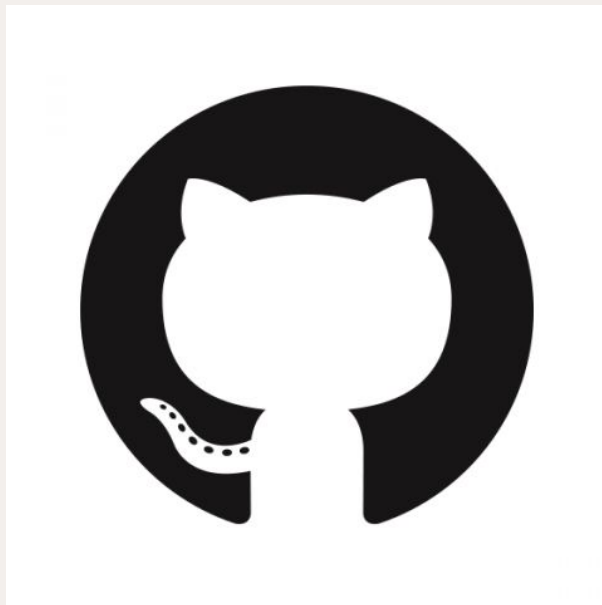
Rep Counter

Reps: 0  
Angle: 97  
min:25.7 max:132.4  
up:57.7 down:100.4

PROBLEMS

(.venv) PS C: ...  
REP! reps=1,  
REP! reps=2,  
(.venv) PS C: ...  
VIDEO\_PATH: C: ...  
Exists?: True  
INFO: Created  
WARNING: All  
W0000 00:00:1  
ck tensors.  
W0000 00:00:1  
ck tensors.  
W0000 00:00:1  
DIMENSIONS or

python



Finally I used those landmarks + angle thresholds to implement a rep counter in real time.

Check repo: ***[https://github.com/jordanaftali/rep\\_count-ml](https://github.com/jordanaftali/rep_count-ml)***