

YOGAPAL



Priyansh Desai – Rushiraj Gadhvi – Mayank Mor

problem statement

Despite the growing popularity of yoga, many people cannot fully benefit from it due to incorrect postures, often requiring supervision or mentoring.

With everyone being busy with their schedule, makes it even more challenging to find such guidance , leading to suboptimal or discontinued yoga practice.

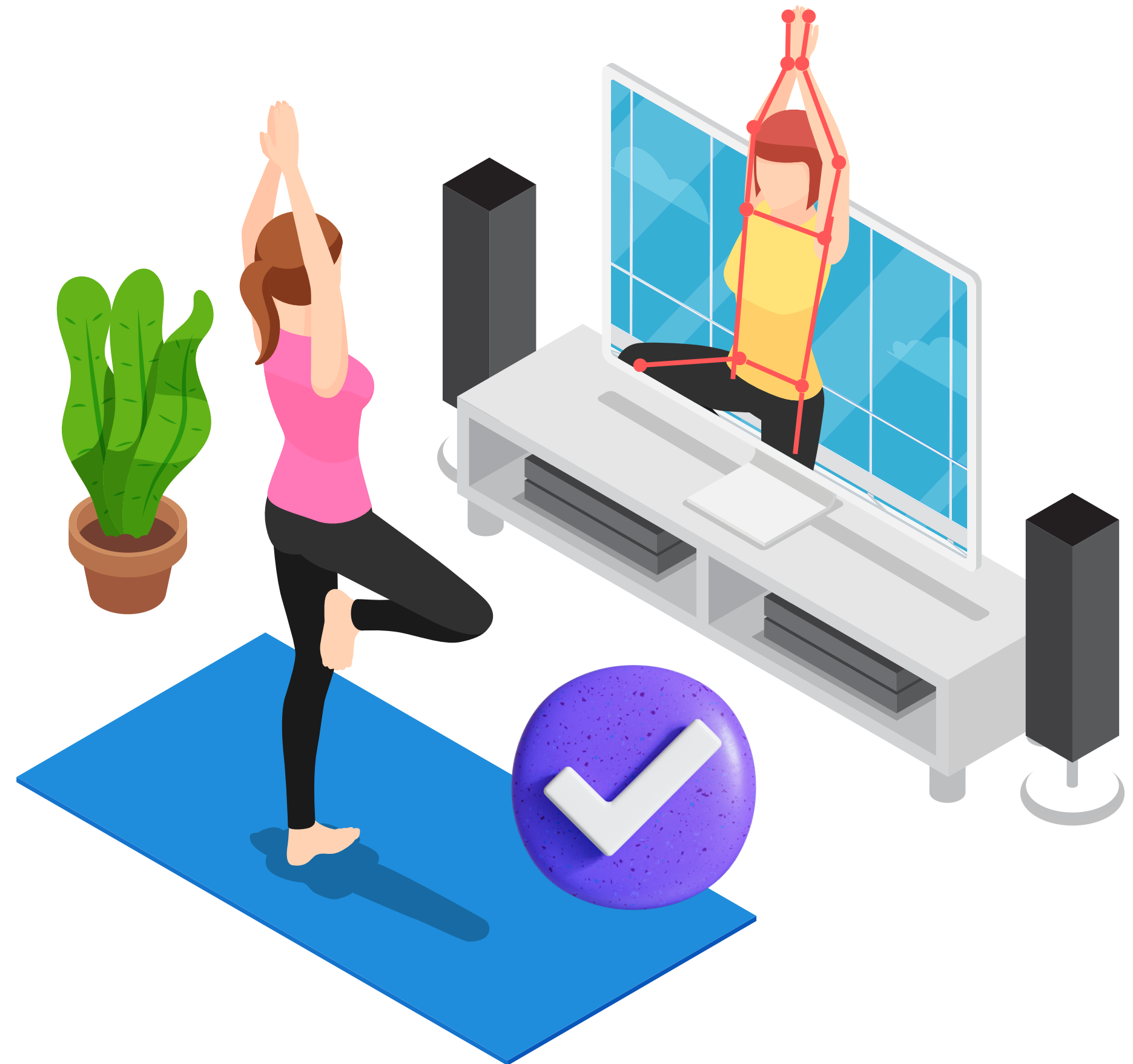
A solution is needed to **enable independent and correct yoga practice for maximum effectiveness.**



vision

Develop a yoga pose recognition and correction suggestion model aimed at promoting proper yoga practice for maximum effectiveness of asanas.

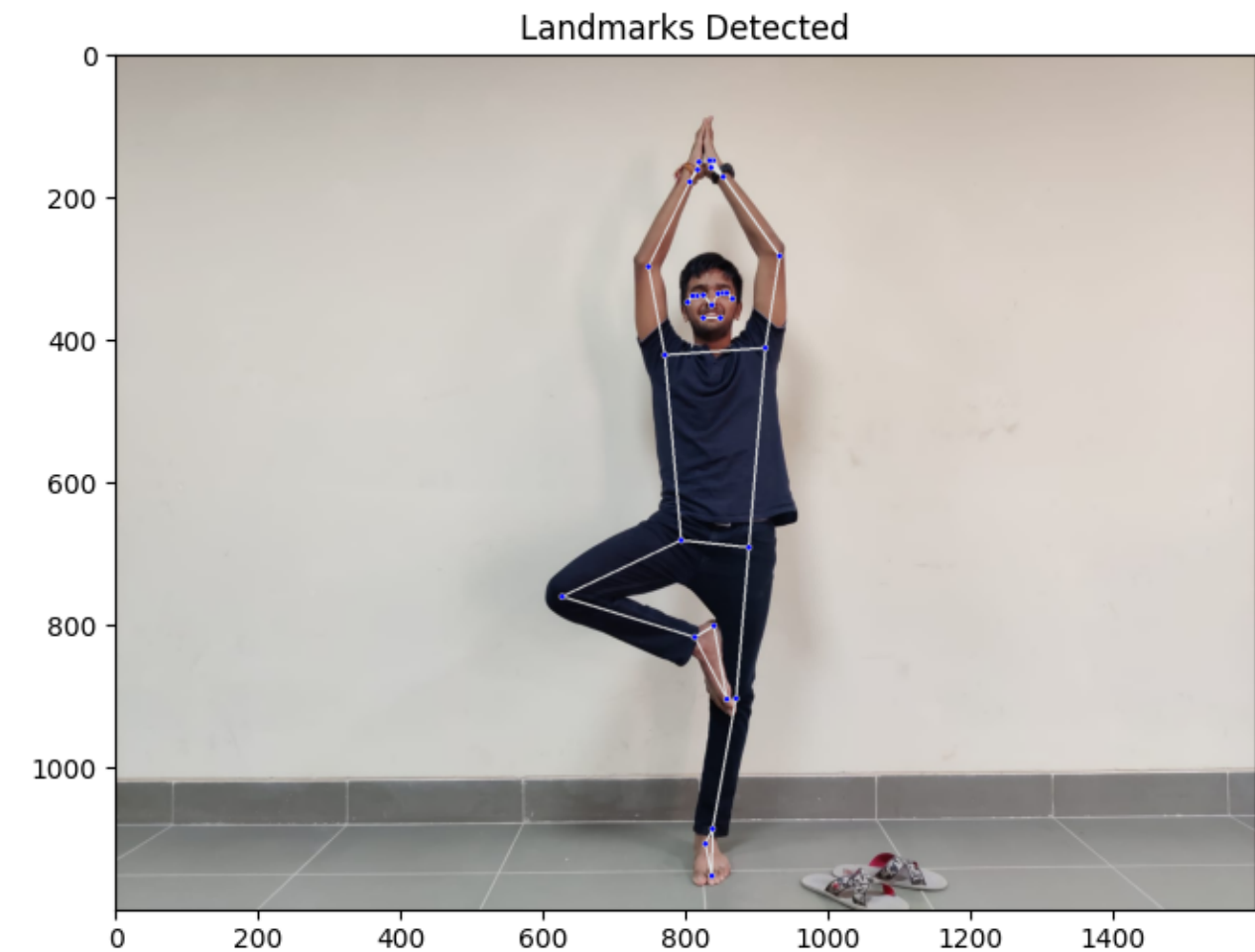
This model will analyze a user's yoga pose, provide **feedback**, and offer suggestions to **help individuals perform yoga correctly**.



midterm recap

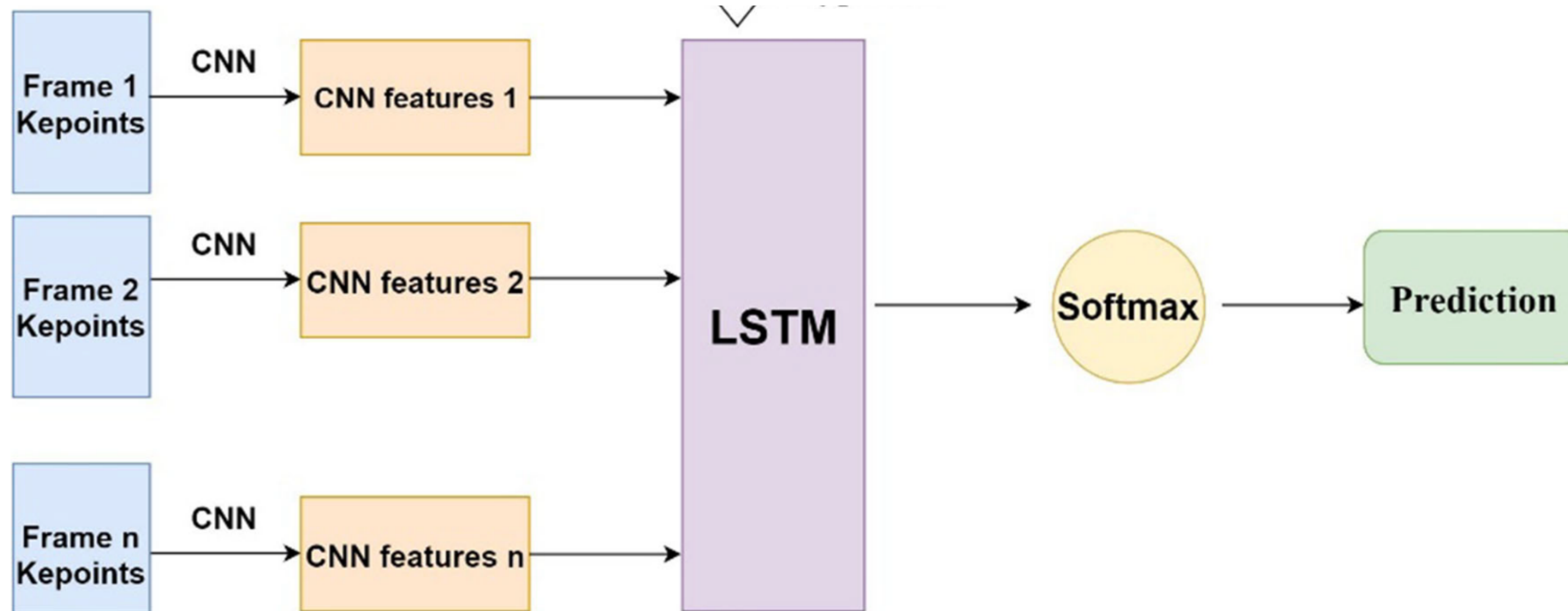
Prediction on static yoga pose images.

	precision	recall	f1-score	support
tree	1.00	0.91	0.95	33
cobra	0.91	1.00	0.95	61
downdog_data	0.96	0.98	0.97	45
goddess	1.00	1.00	1.00	20
warrior	1.00	0.84	0.91	19
chair	0.95	0.91	0.93	23
accuracy			0.96	201
macro avg	0.97	0.94	0.95	201
weighted avg	0.96	0.96	0.95	201



Literature Review: Real-Time Yoga Recognition

This Paper also utilised a hybrid model of CNN and **LSTM**



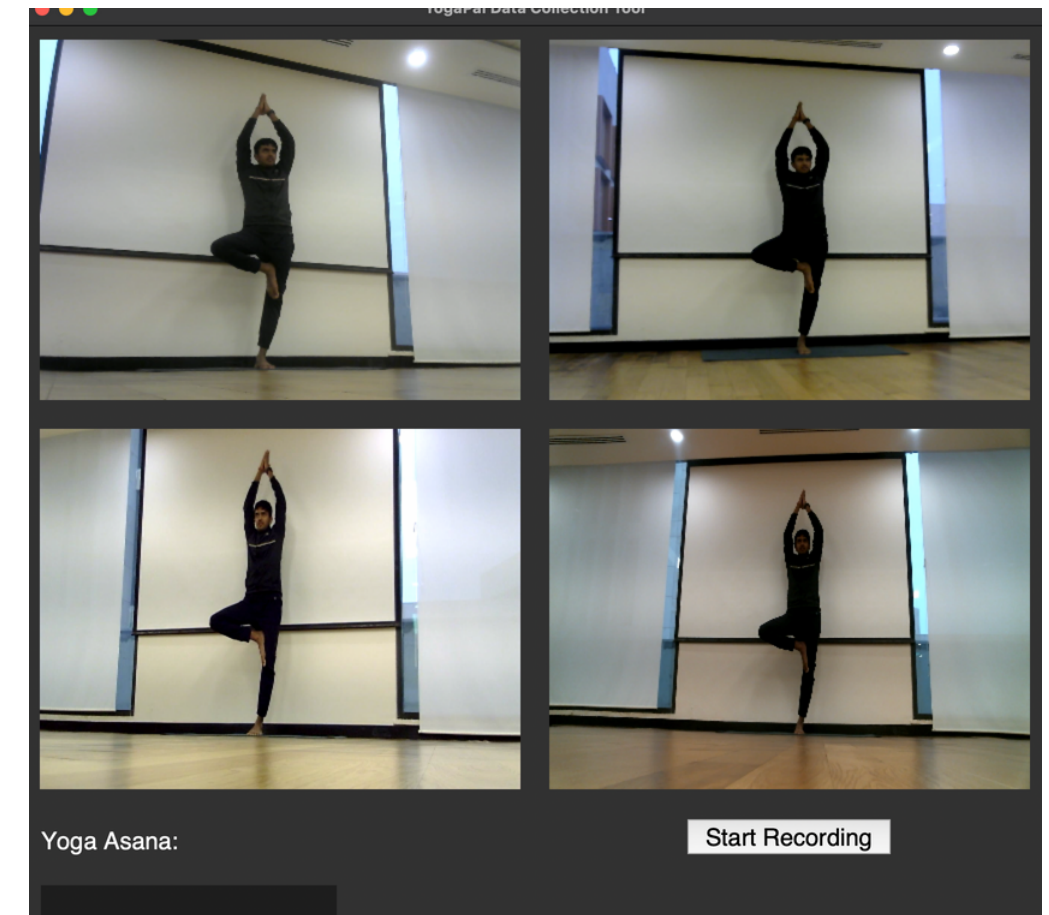
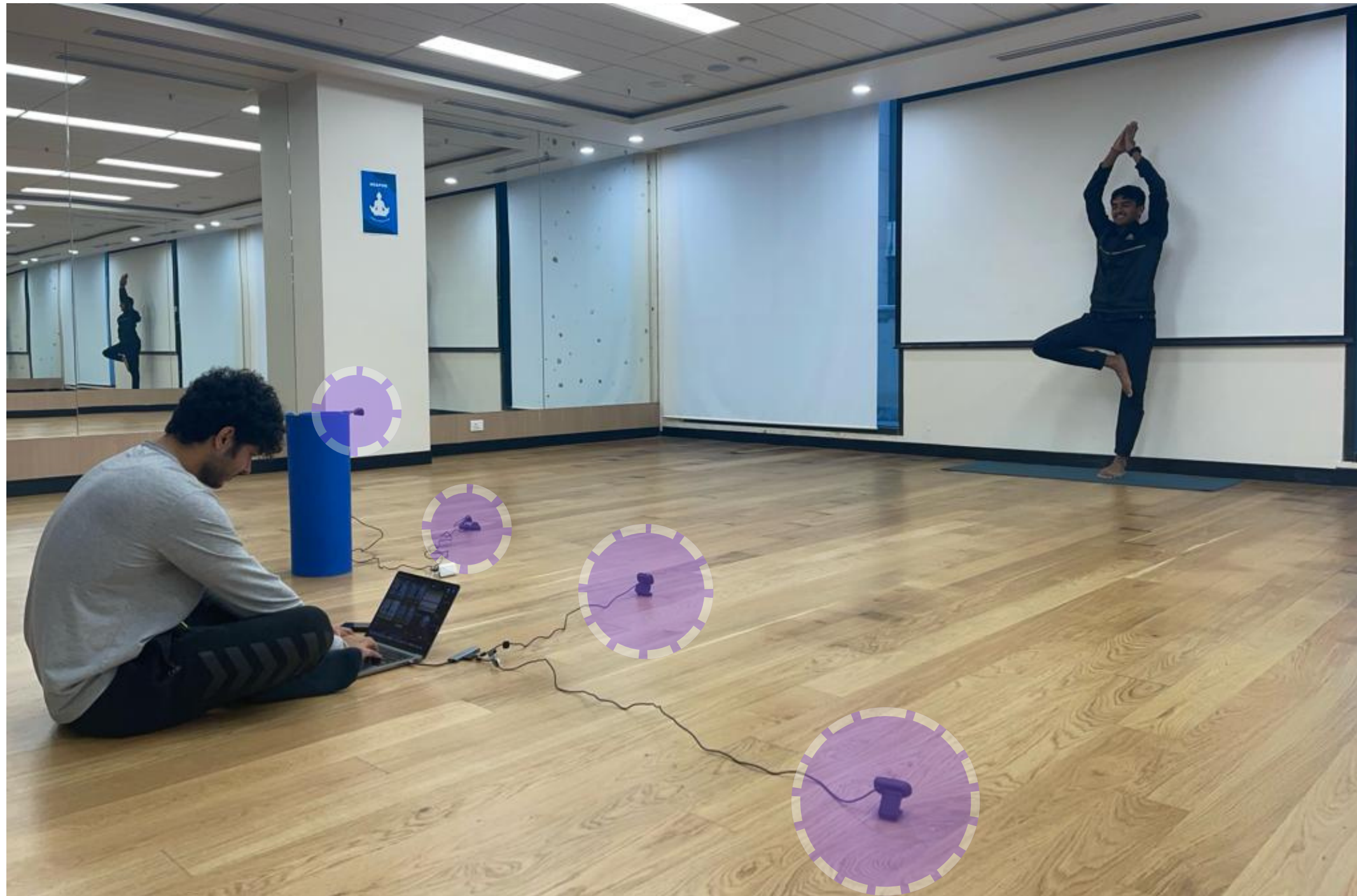
Dataset: created their own (88 vid)
15 individuals

full video pass
Accuracy: 98.92%

shifting to videos

0.9996339082717900	0.43132367730140700	0.2612672448158260	-0.1999279409646990	0.999569296836853	0.44605010747909500	0.2501174807548520	-0.14360958337783800	0.9993972182273870	0.42991337180137600	0.2571716606616970	-0.1367526799440380	0.9990891814231870	0.4419882595539090	0.2714942395687100	-0.17781034111976600	0.9998248815536500	0.43572086095809900
0.9996684789657590	0.4313212037086490	0.2618178725242620	-0.19578680396080000	0.9996103644371030	0.4460439085960390	0.25029224157333400	-0.14034387469291700	0.9994543194770810	0.4298829734325410	0.2574848532676700	-0.1347151664876940	0.9991770386695860	0.4420033395290380	0.27166086435318000	-0.17362871766090400	0.9998416304588320	0.43573078513145400
0.9996994733810430	0.43132713437080400	0.26242688298225400	-0.19806316494941700	0.9996471405029300	0.44607269763946500	0.25063788890838600	-0.14224623143672900	0.9995055794715880	0.4298876225948330	0.25798559188842800	-0.13651612401008600	0.999255895614624	0.44209620356559800	0.2723190188407900	-0.17557142674922900	0.9998565316200260	0.43578797578811600
0.999727725982666	0.4315688908100130	0.2632118761539460	-0.20106495916843400	0.9996806383132940	0.4464186429977420	0.25116199254989600	-0.1432662010192870	0.9995524287223820	0.4299979507923130	0.2585945129394530	-0.13936592638492600	0.9993276000022890	0.44270583987236000	0.27320176362991300	-0.17746640741825100	0.9998701810836790	0.4363097846508030
0.9997524619102480	0.4322623908519750	0.26305416226387000	-0.20308224856853500	0.9997101426124570	0.44710320234298700	0.25114932656288100	-0.14525577425956700	0.9995932579040530	0.43035122752189600	0.25856438279151900	-0.14266984164714800	0.9993911981582640	0.4436138868331910	0.2731837034225460	-0.17918889224529300	0.9998820424079900	0.43721404671669000
0.9997745156288150	0.43306905031204200	0.2630826234817510	-0.19537274539470700	0.999736487865448	0.44798949360847500	0.25125691294670100	-0.13818487524986300	0.9996297955513	0.4308112859725950	0.2585563063621520	-0.1355685442686080	0.9994481801986690	0.4447174072265630	0.2734736204147340	-0.17190435528755200	0.9998926520347600	0.4382244944572500
0.9997949004173280	0.43422237038612400	0.26324883103370700	-0.1999741643667220	0.9997607469558720	0.449155330657959	0.25159594416618300	-0.1433832198381420	0.999663770198822	0.43169543147087100	0.25858914852142300	-0.1400982290506360	0.9995002746582030	0.44586023688316300	0.2739924192428590	-0.17646244168281600	0.9999023675918580	0.4394344091415410
0.9998120665550230	0.435318261384964	0.26317650079727200	-0.18531349301338200	0.9997814297676090	0.4499475657939910	0.2516404390335080	-0.13115015625953700	0.9996923208236690	0.43296194076538100	0.2585308253765110	-0.12680692970752700	0.9995450377464290	0.4465924799442290	0.2739773392677310	-0.1628132164478300	0.9999106526374820	0.440310001373291
0.9998279809951780	0.43636322021484400	0.2630777359008790	-0.18728052079677600	0.999800443649292	0.4509738087654110	0.2516395151615140	-0.1337868571281430	0.9997187852859500	0.43420684337616000	0.25829723477363600	-0.1282186657190320	0.9995861053466800	0.4474053680896760	0.27394917607307400	-0.16503530740737900	0.9999183416366580	0.44125232100486800
0.9998432397842410	0.43739229440689100	0.26305338740348800	-0.1933261901140210	0.9998184442520140	0.45196154713630700	0.2516423761844640	-0.1384669840335850	0.9997438192367550	0.4354926645755770	0.2581537365913390	-0.1333494633436200	0.9996244311332700	0.44825851917266800	0.2739369869232180	-0.17074590921402000	0.9999256134033200	0.442252814769745
0.9998558759689330	0.4394141733646390	0.262566477060318	-0.19452132284641300	0.9998335838317870	0.45344728231430100	0.2515150010585790	-0.14057500660419500	0.9997645616531370	0.43740731477737400	0.257531076669693	-0.1354091465473180	0.999657154083252	0.4498218894004820	0.27380141615867600	-0.17215770483017000	0.9999316334724430	0.44417861104011500
0.9998677372932430	0.44163572788238500	0.2620467245578770	-0.19386793673038500	0.9998477697372440	0.45564159750938400	0.2514999806880950	-0.14083440601825700	0.9997840523719790	0.43934619426727300	0.25717177987098700	-0.13527701795101200	0.9996874928474430	0.451748251914978	0.2737829089164730	-0.17148754000663800	0.9999372363090520	0.44624730944633500
0.9998772740364080	0.4441184103488920	0.26097264885902400	-0.19492347538471200	0.9998593330383300	0.458275705575943	0.25115081667900100	-0.14154958724975600	0.9997994899749760	0.4417192041873930	0.25678330659866300	-0.13610073924064600	0.9997127652168270	0.45428261160850500	0.2735672891139980	-0.1726219356060030	0.9999415874481200	0.44879207015037500
0.9998854398727420	0.445953905582428	0.25989672541618300	-0.18293137848377200	0.9998694062232970	0.46017029881477400	0.25075265765190100	-0.1314205527305600	0.9998116493225100	0.44417300820350600	0.2561657130718230	-0.12547919154167200	0.9997346997261050	0.45605096220970200	0.27308496832847600	-0.16207446157932300	0.9999449849128720	0.45056095719337500
0.9998921155929570	0.4490157961845400	0.25834259390831000	-0.1832202970981600	0.9998778104782100	0.46278512477874800	0.2497348040342330	-0.13042257726192500	0.9998201131820680	0.44719669222831700	0.2549459934234620	-0.12646549940109300	0.9997532367706300	0.45871713757515000	0.2721608877182010	-0.16204829514026600	0.9999474883079530	0.45340022444725000
0.99989820164527890	0.4515780508519200	0.2575940787792210	-0.18679821461105300	0.9998853802680970	0.46570804715156600	0.24919314682483700	-0.13341344892978700	0.9998281202044680	0.45001477003097500	0.25411346554756200	-0.12903088331222500	0.9997693300247190	0.46127188205719000	0.2716825604438780	-0.16516472399243800	0.9999498128899090	0.45596185326576200
0.9999042749404910	0.4548332095146180	0.2568379342556	-0.19318832457065600	0.9998930096626280	0.4694043695926670	0.2486371397972110	-0.1405496746301650	0.9998382329940800	0.45340216159820600	0.2531890869140630	-0.13438670337200200	0.9997851848602300	0.4645749032497410	0.2710067331790920	-0.17198100686073300	0.9999525547027590	0.45938071608543400
0.9999088644981380	0.45767703652381900	0.25636017322540300	-0.19270560145378100	0.9998988509178160	0.47198542952537500	0.24852074682712600	-0.1405150145292280	0.9998461604118350	0.45637306571006800	0.25248003005981400	-0.1332167237997060	0.9997972846031190	0.46710941195488000	0.2707500457763670	-0.17164653539657600	0.999954879283905	0.46188923716545100
0.9999123215675350	0.46056678891181900	0.25543949007988000	-0.19044318795204200	0.9999033808708190	0.4746147096157070	0.24843212962150600	-0.13882075250148800	0.9998513460159300	0.45943349599838300	0.2517267167568210	-0.13152259588241600	0.9998069405555730	0.46966248750686600	0.2702350616455080	-0.16961544752121000	0.999956488609314	0.464388906955719
0.9999159574508670	0.4642586410045620	0.25533849000930800	-0.18883885443210600	0.9999081492424010	0.4783920347690580	0.24844442307949100	-0.13554856181144700	0.9998565316200260	0.4625566601753240	0.2514294683933260	-0.13098162412643400	0.9998178482055660	0.47322046756744400	0.2701970934867860	-0.1668885814189910	0.9999579787254330	0.4681027829647060
0.9999215602874760	0.4672495722770690	0.2553319036960600	-0.2092929631471630	0.9999144673347470	0.48157885670661900	0.2485121786594390	-0.1523500233888630	0.9998661279678350	0.4650849997997280	0.25131088495254500	-0.15134620666503900	0.9998316168785100	0.4762977659702300	0.2702394723892210	-0.18488739430904400	0.9999606609344480	0.471166729927063
0.9999269843101500	0.46999678015708900	0.2553844451904300	-0.207809716463089	0.9999205470085140	0.48389536142349200	0.24870441854000100	-0.15047802031040200	0.9998753070831300	0.46733197569847100	0.2513425052165990	-0.14924798905849500	0.9998446106910710	0.4789705276489260	0.27051371335983300	-0.18343539535999300	0.9999632835388180	0.4737953245639800
0.9999309182167050	0.47220680117607100	0.25545674562454200	-0.2056570053100590	0.9999250769615170	0.4859049916267400	0.24885858595371200	-0.14395925402641300	0.9998823404312130	0.46969273686409000	0.25123465061187700	-0.14806441962719000	0.9998549818992620	0.48088663816452000	0.27052217721939100	-0.17861057817935900	0.9999653100967410	0.4757841229438780
0.9999348521232610	0.47361212968826300	0.2555040121078490	-0.2072163224220280	0.9999295473098760	0.487569123506546	0.2489108294248580	-0.14917023479938500	0.9998895525932310	0.4715960621833800	0.2510780692100530	-0.14905385673046100	0.9998651146888730	0.482115238904953	0.2704619765281680	-0.18221203982830000	0.9999674558639530	0.47694429755210900
0.9999366402626040	0.47484368085861200	0.2545762360095980	-0.20526880025863600	0.999931812286377	0.488552490234400	0.249127814769750	-0.14807294309139	0.9998934268951	0.47319361567497300	0.25046210086210	-0.146790168808000	0.9998713731765750	0.4832012355327610	0.27024295926094100	-0.18094991147518200	0.9999688683754270	0.47790876030921900
0.999940037727356	0.4760287404060360	0.25446662306785600	-0.22309510409832	0.9999355673789980	0.490014842101000	0.249066282010000	-0.164252087000000	0.999900000000000	0.472000000000000	0.250000000000000	-0.142470000000000	0.9998800754547120	0.4844819903373720	0.2701803147792820	-0.19734913110733000	0.9999707937240600	0.4790043532848360
0.9999438524246220	0.47770896553993200	0.2543119788169860	-0.2319067269563680	0.9999396204948	0.49003996913	0.249038934174650	-0.1726120798830	0.999907000000000	0.470096400000000	0.250000000000000	-0.129660000000000	0.999890161514280	0.48597171902656600	0.27016109228134200	-0.20601657032966600	0.9999728202819820	0.48061344027519200
0.999948263168335	0.4788183271884920	0.25423765182495100	-0.22363968193531000	0.9999443292617	0.490085000000000	0.249043900000000	-0.164650000000000	0.999914000000000	0.470000000000000	0.250000000000000	-0.129660000000000	0.99989374080658	0.4869783818721770	0.27015864849090600	-0.19814787805080400	0.9999749660491940	0.4814373254776000
0.9999521970748900	0.4794235825538640	0.25328460335731500	-0.21096840500831600	0.9999485611915590	0.490092000000000	0.249081000000000	-0.149116200000000	0.999920000000000	0.470000000000000	0.250000000000000	-0.129660000000000	0.999896837900000	0.486919000000000	0.2700130045413970	-0.1854144185781480	0.9999769926071170	0.4820362329483030
0.9999560713768010	0.47915592789650000	0.25251707434654200	-0.209075033664														

data collection



8 Individuals ×
6 Asana Poses ×
4 Camera Angles ×

192 Videos

32 Videos per Asana

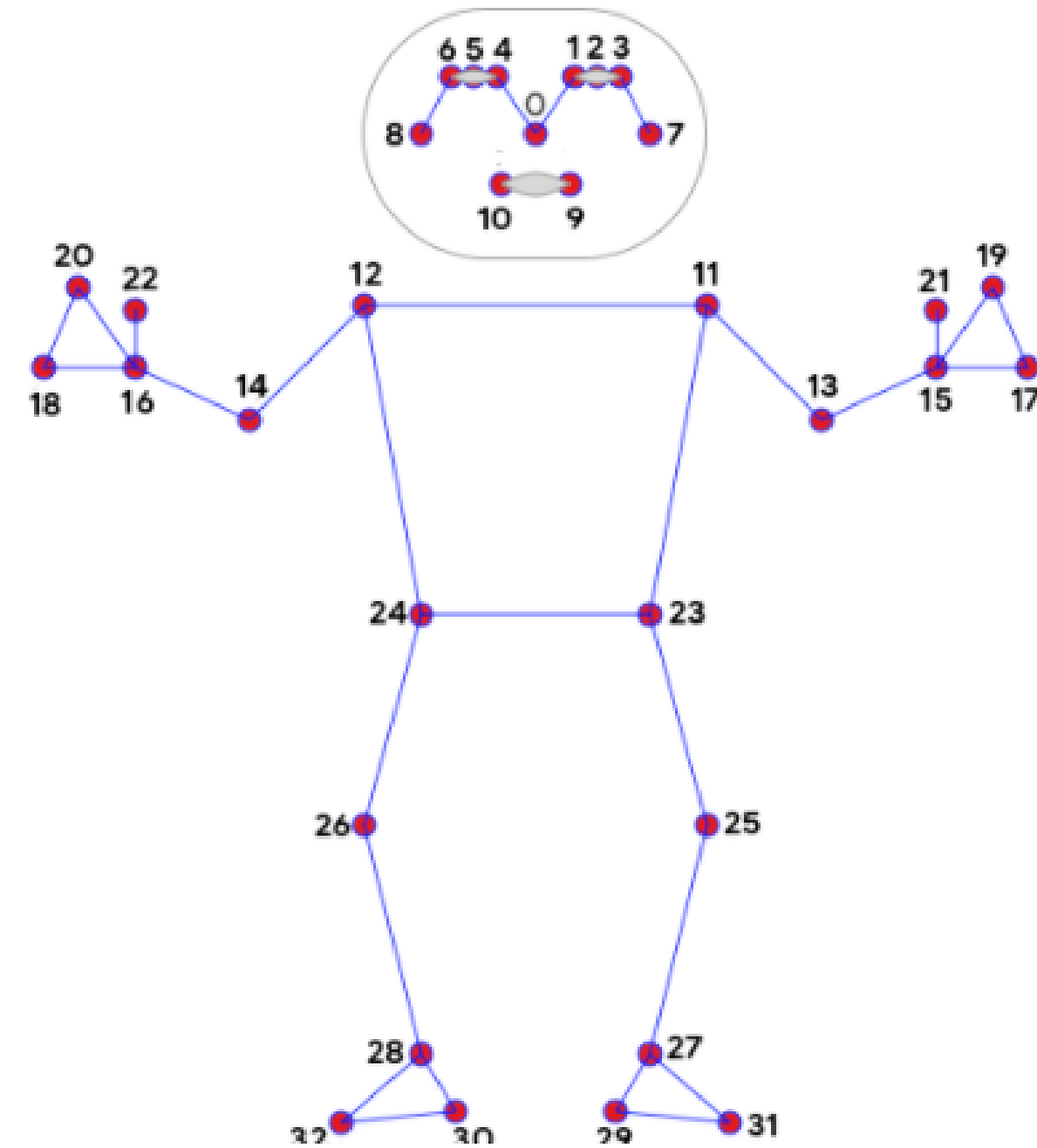
Note: cameras had different frame rates

extracting data

We used MediaPipe's opensource body pose estimation library that uses deep learning models to detect and track key landmarks on the human body in real-time from images or videos.

Mediapipe can detect **33 Landmarks** from a input body pose image.

We extracted landmark data for all the frames in the dataset.



- | | | | |
|--------------------|--------------------|-----------------|----------------------|
| 0. nose | 9. mouth_left | 17. left_pinky | 25. left_knee |
| 1. left_eye_inner | 10. mouth_right | 18. right_pinky | 26. right_knee |
| 2. left_eye | 11. left_shoulder | 19. left_index | 27. left_ankle |
| 3. left_eye_outer | 12. right_shoulder | 20. right_index | 28. right_ankle |
| 4. right_eye_inner | 13. left_elbow | 21. left_thumb | 29. left_heel |
| 5. right_eye | 14. right_elbow | 22. right_thumb | 30. right_heel |
| 6. right_eye_outer | 15. left_wrist | 23. left_hip | 31. left_foot_index |
| 7. left_ear | 16. right_wrist | 24. right_hip | 32. right_foot_index |
| 8. right_ear | | | |

feature engineering



Feature Extraction from Data: Computing the angles between these selected joints using trigonometric functions. For instance, you can calculate the angle between the line connecting the shoulder and elbow and the line connecting the elbow and wrist.

Total Angle possible:

$(33 - 16)C3 \rightarrow$ **680 different angles**



picking best frames

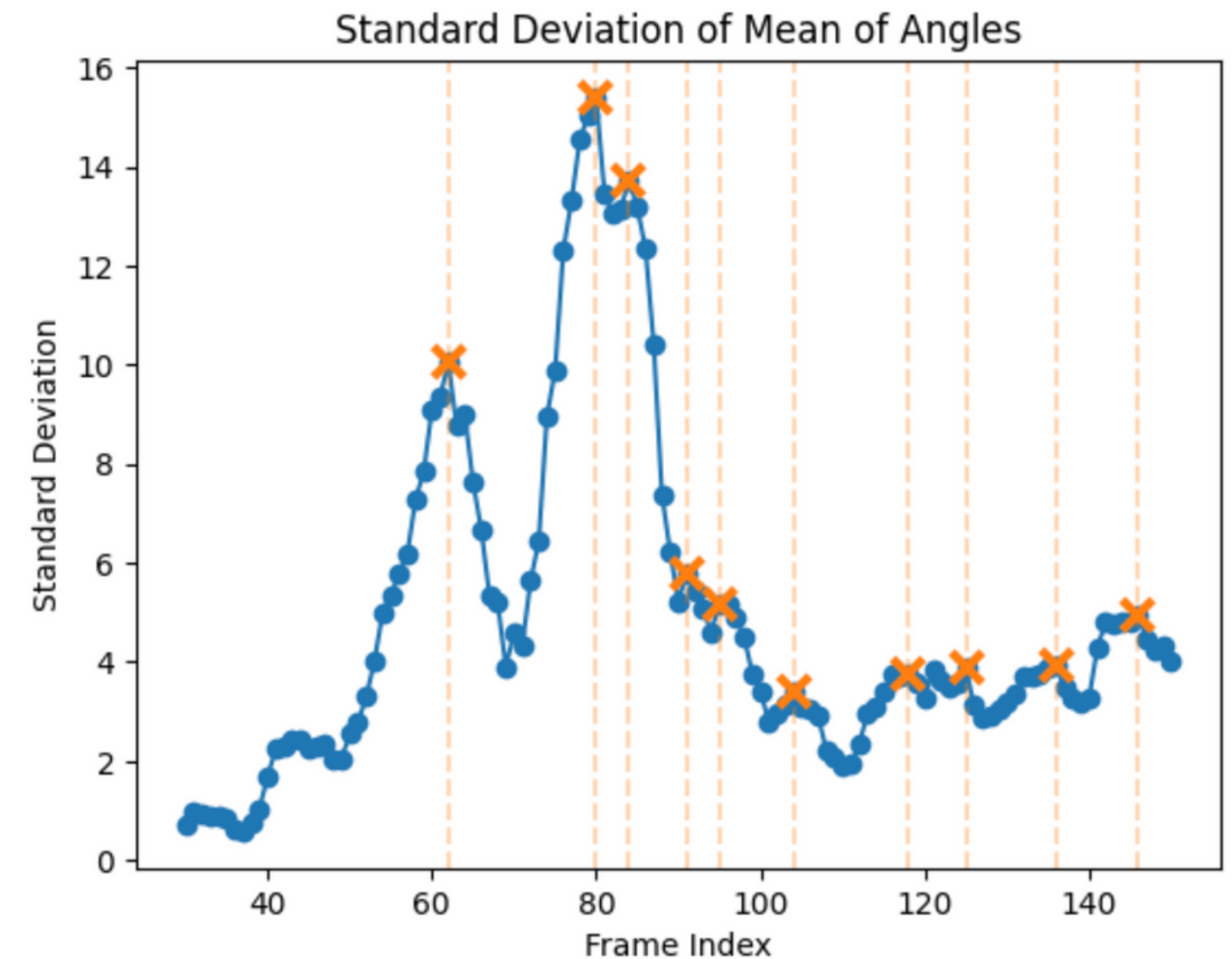
	f1	f2	f3	f4	f5	f6	f7
0	60.169878	65.454222	29.440543	22.171905	139.699184	130.076897	169.435381
1	61.738340	66.627413	29.599132	21.977421	137.388626	127.934486	168.690023
2	50.314153	63.371661	19.328409	12.280972	157.002997	160.252816	172.979298
3	57.588088	66.729180	29.000828	20.832816	141.045193	135.178421	168.950837
4	60.774958	68.051661	28.150639	19.533750	139.581088	133.122874	168.542387

Videos on average had roughly about ~**270 frames**

↓
REDUNDANT INFORMATION

↓
Selecting important frames

↓
calculated standard deviation
for **n**th timestamp with a
range of **(n-2) to (n+2)**



↘ **scipy.signal.find_peaks** ↗

find local maxima

frames corresponding to peaks

Top Frames



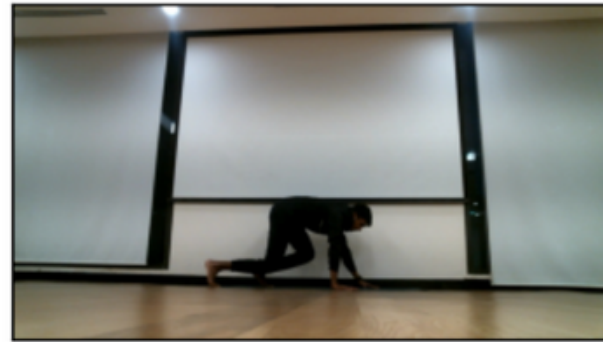
Frame 1



Frame 2



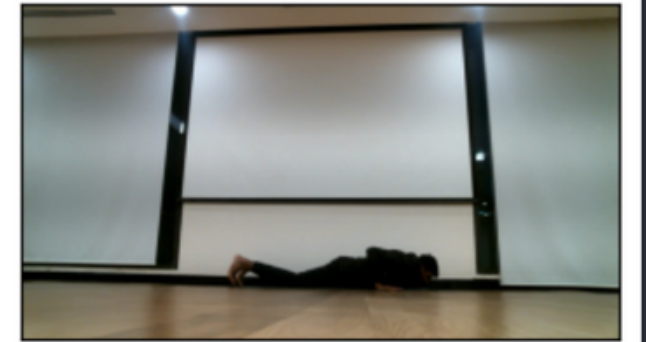
Frame 3



Frame 4



Frame 5



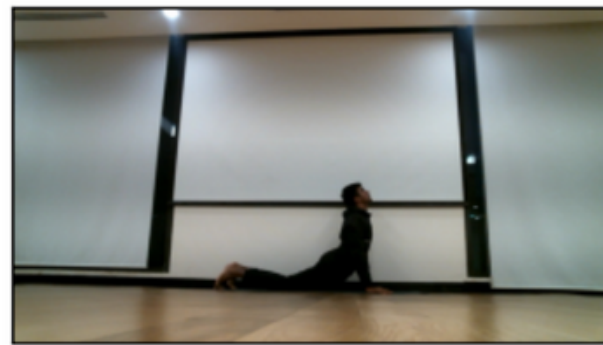
Frame 6



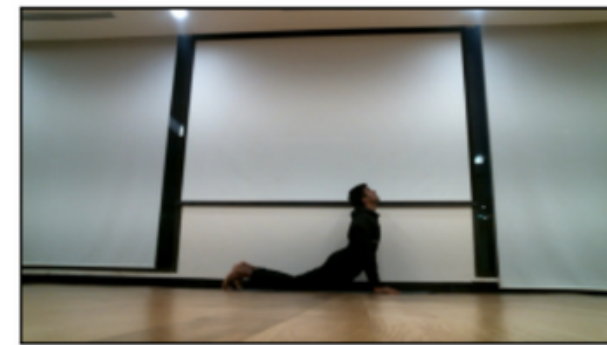
Frame 7



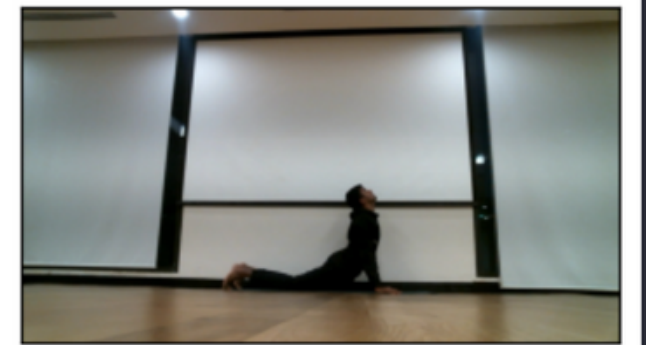
Frame 8



Frame 9

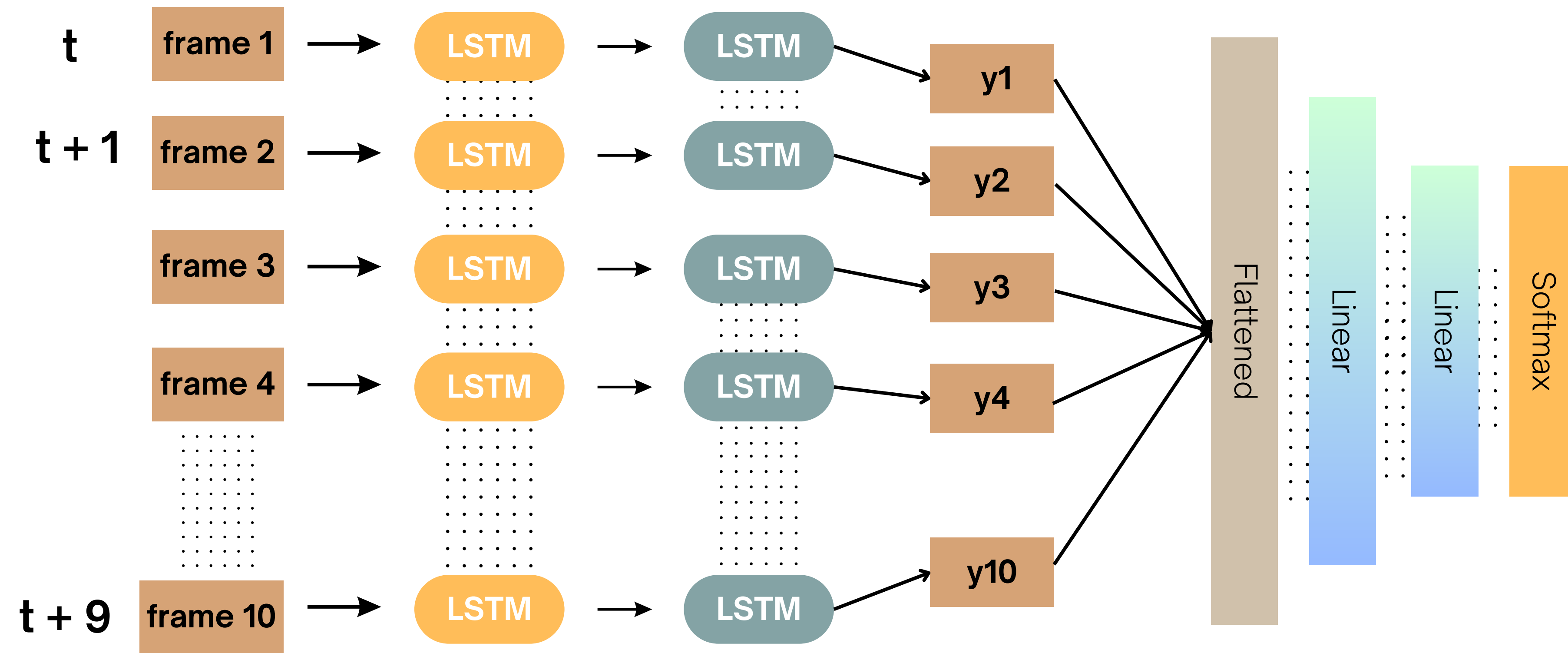


Frame 10



classification model

model

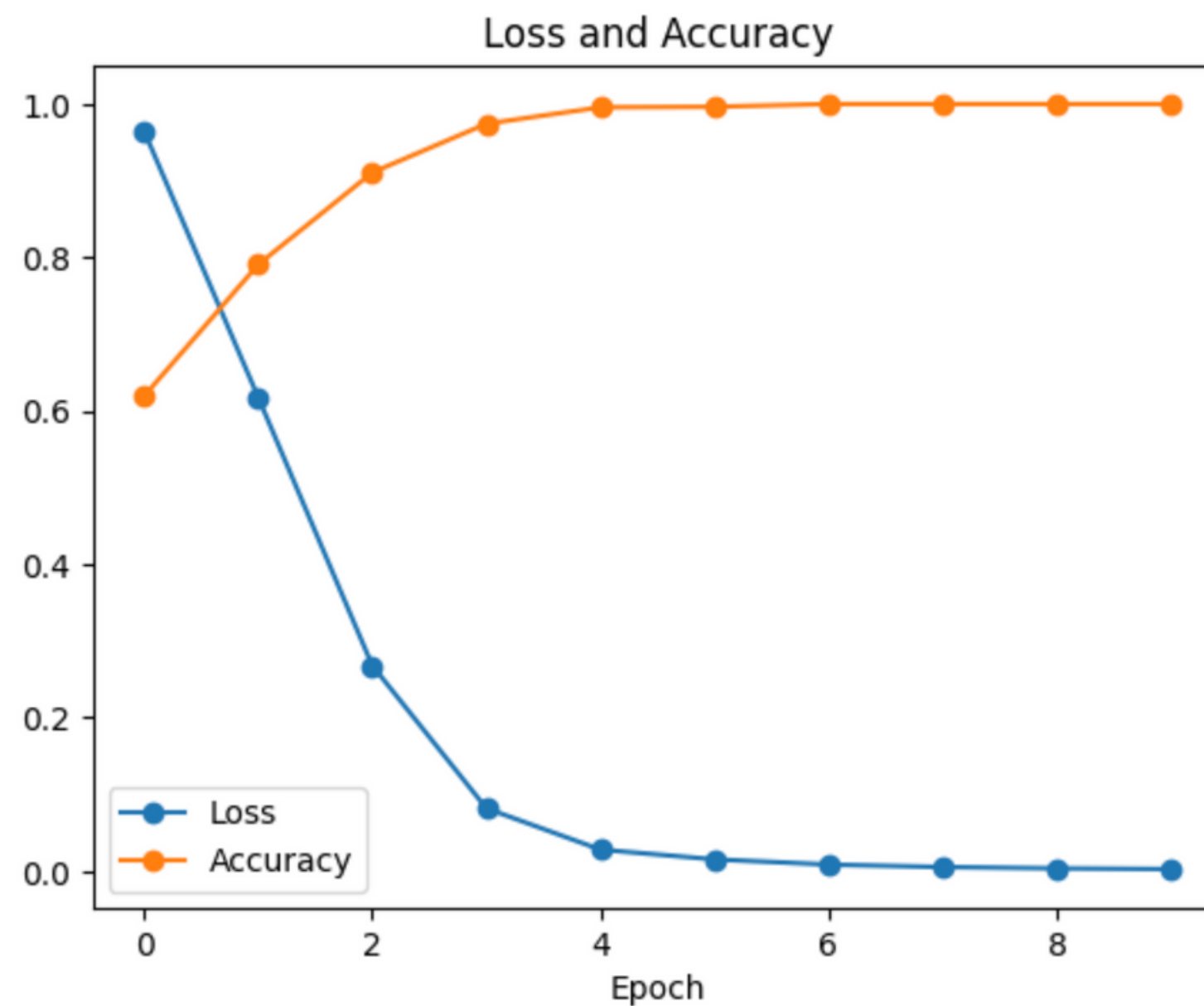


Long Short Term Memory

- We designed an LSTM (Long Short-Term Memory) neural network to classify six distinct yoga poses. The model was fed with the most significant frames from each video, selectively choosing the top 10 frames to capture the essence of the movements.
- Our training was robust, consisting of 176 videos, translating to 1,760 individual frames. These frames were batched in sets of 10 and sequentially fed into our LSTM model, allowing for a comprehensive understanding of temporal dynamics within each pose.
- The LSTM model demonstrated exceptional performance, achieving a high classification accuracy of 95%. This indicates a strong predictive ability to discern among the six yoga poses, validating the effectiveness of our frame selection strategy and LSTM architecture.

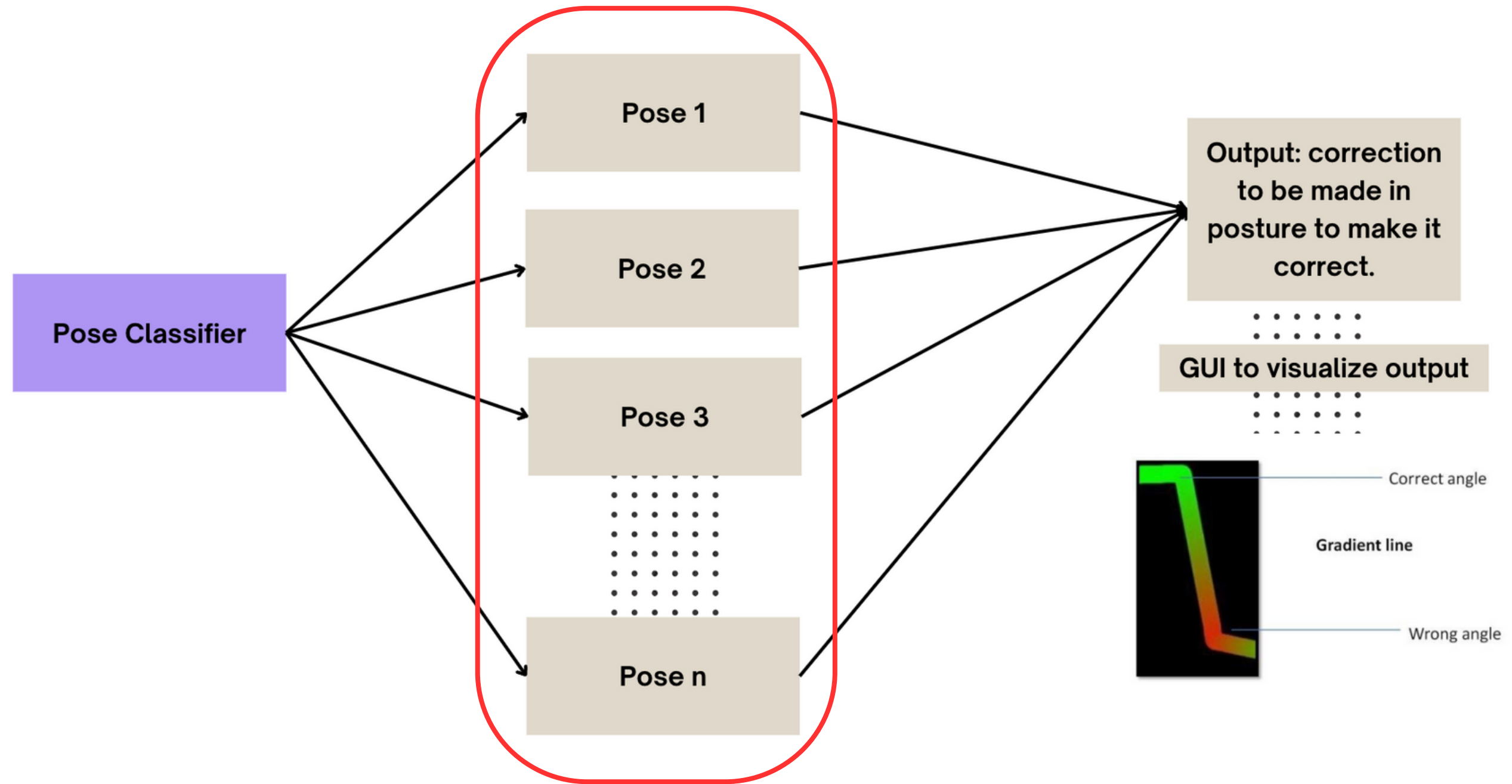
performance

	precision	recall	f1-score
0.0	1.00	1.00	1.00
1.0	1.00	0.98	0.99
2.0	0.92	0.96	0.94
3.0	0.94	0.96	0.95
4.0	1.00	1.00	1.00
5.0	0.97	0.94	0.96
accuracy			0.97
macro avg	0.97	0.97	0.97
weighted avg	0.97	0.97	0.97

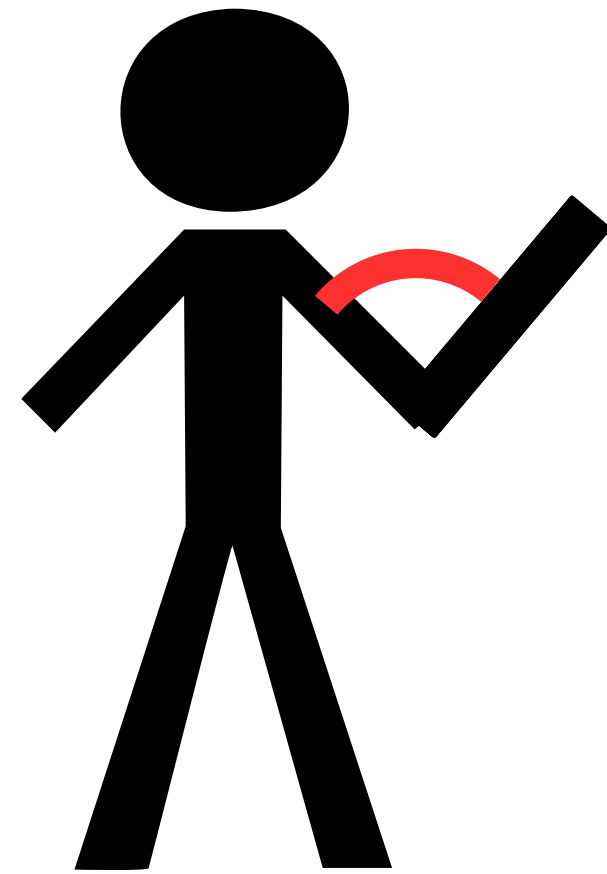


correction model

correction models

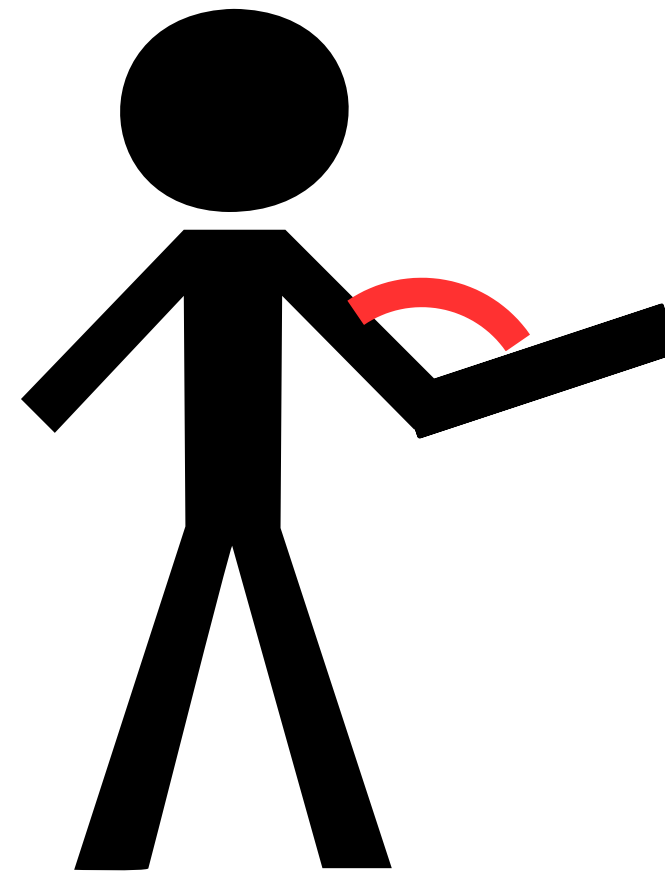


features for correction model



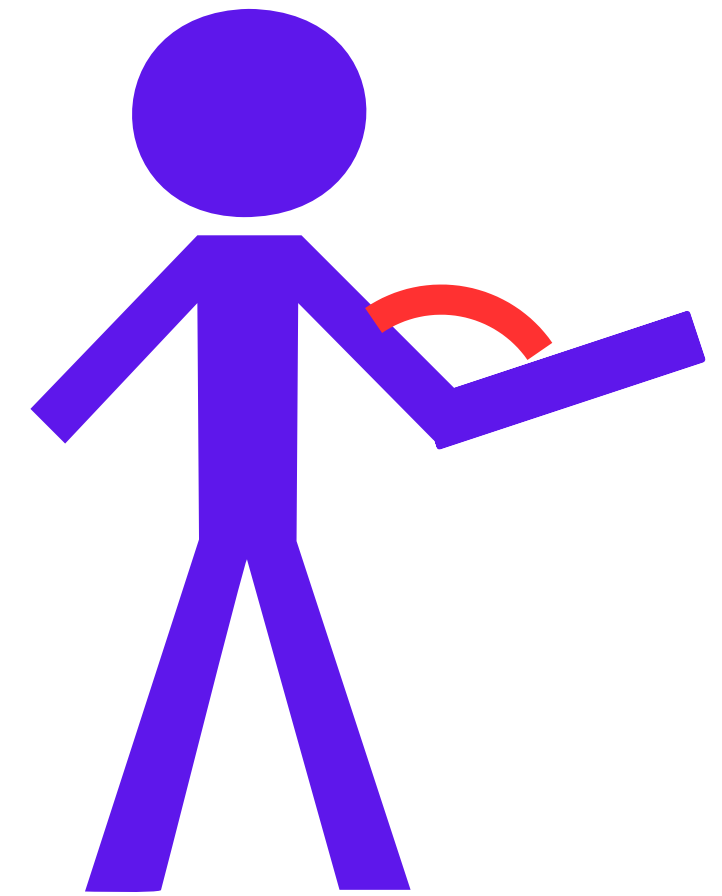
user @ t

CORRECTION
MODEL



predicted @ $t + 1$

ERROR between angles



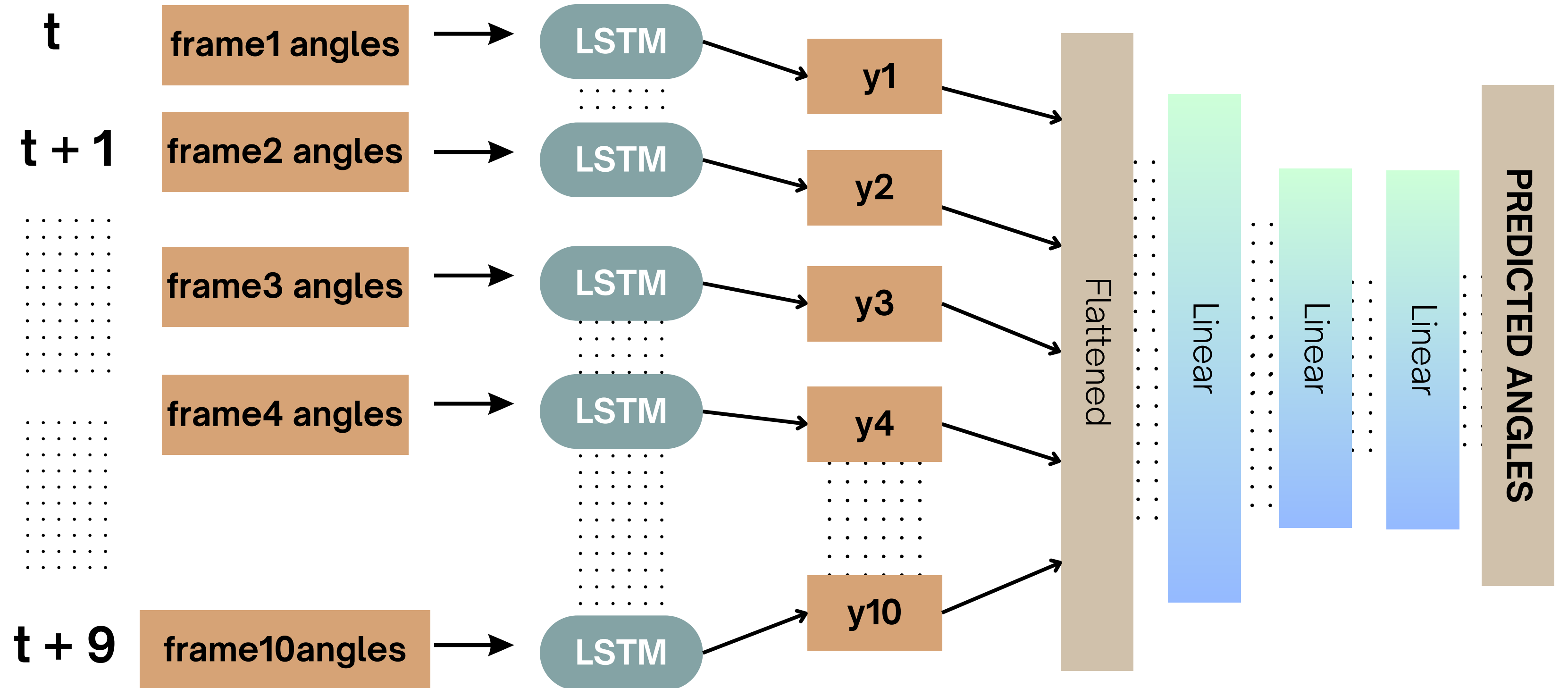
user @ $t + 1$

forecasting angles from **[t]**th to **[t+1]**th frame

features (knowledge based)

```
# '''  
#     f1: angle between left shoulder, left elbow, left wrist  
#     f2: angle between right shoulder, right elbow, right wrist  
#     f3: angle between left shoulder, left hip, left knee  
#     f4: angle between right shoulder, right hip, right knee  
#     f5: angle between left hip, left knee, left ankle  
#     f6: angle between right hip, right knee, right ankle  
#     f7: angle between nose, left shoulder, left hip  
#     f8: angle between nose, right shoulder, right hip  
#     f9: angle between left shoulder, nose, right shoulder  
#     f10: angle between left knee, left ankle, left foot index  
#     f11: angle between right knee, right ankle, right foot index  
#     f12: angle between left index, left wrist, left thumb  
#     f13: angle between right index, right wrist, right thumb  
#     f14: angle between left shoulder, left hip, left foot index  
#     f15: angle between right shoulder, right hip, right foot index
```

model

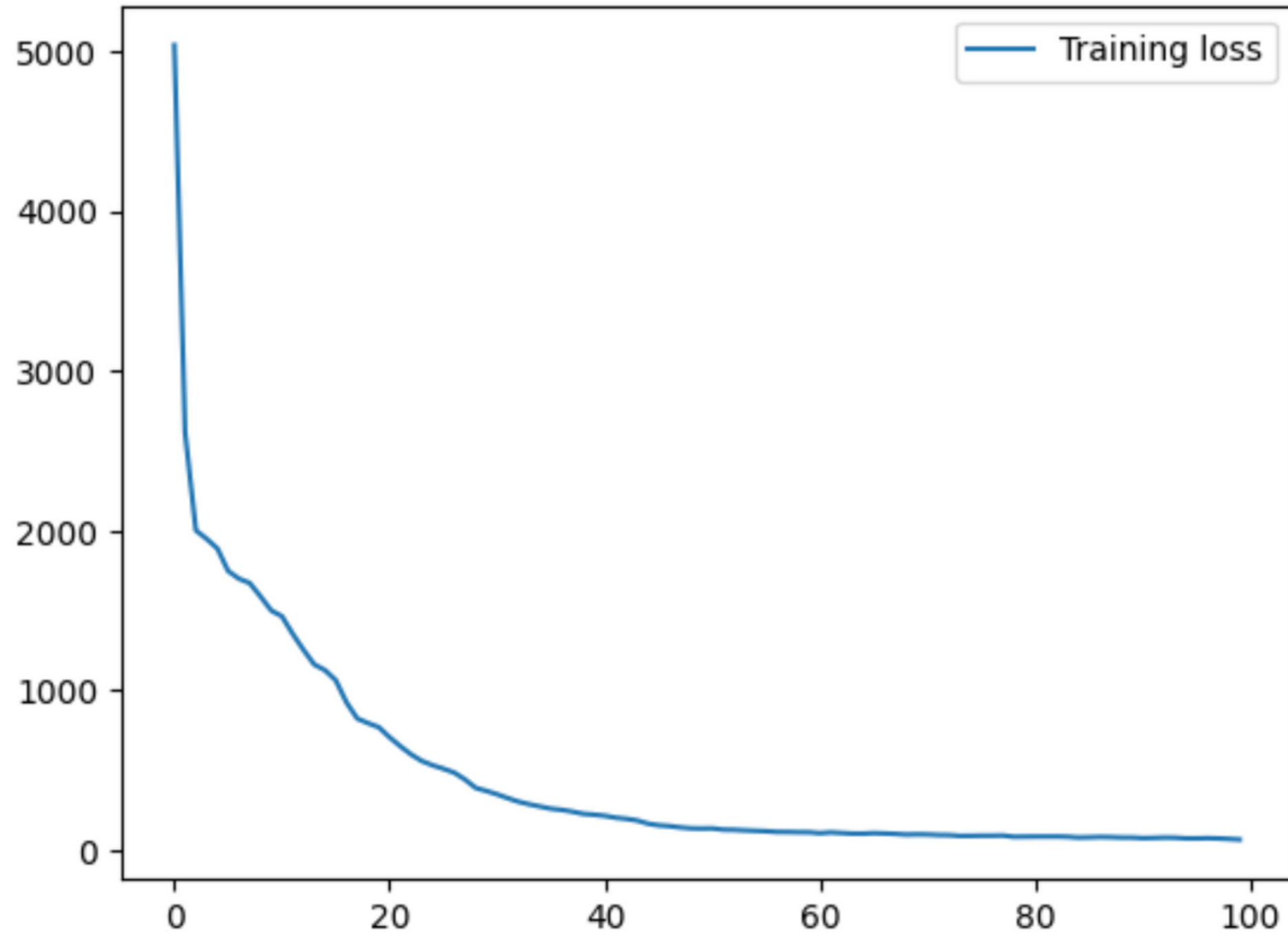


- Our LSTM correction model is trained to anticipate the posture of the subsequent frame (t+1) based on the current frame (t), facilitating real-time pose adjustment.
- The model's prediction for frame (t+1) is juxtaposed with the user's actual pose at frame (t+1) to identify discrepancies.
- By examining 15 key angles, we quantify the deviation between the LSTM's predicted pose and the user's actual pose, enabling us to provide precise feedback for pose improvement.

Data Augmentation

- Faced with a scarcity of data, our initial dataset comprised **only 32 videos for a specific yoga asana**. This limitation necessitated innovative solutions for data expansion.
- To augment our dataset, we implemented a series of transformations on selected images. The code demonstrates the use of random rotations, horizontal flips, resizing, and cropping to generate diverse training samples.
- As a result of these transformations, we successfully expanded our dataset to include **512 videos of augmented data** for each pose. This is achieved by iterating **16 times** over each image, as seen in the loop structure of the code.

```
data_trans = transforms.Compose([
    transforms.PILToTensor(),
    # transforms.Resize(size=480),
    transforms.CenterCrop((800,800)), # Example
    transforms.RandomHorizontalFlip(0.5),
    transforms.RandomRotation((-20, 20)),
    transforms.ToPILImage(),
])
```



DATA PROBLEM ???

**statistical
threshold to be set
for suggestion**



```
[ 8.597221  8.819668 10.279508  4.843954  3.7453132  8.4125595  
6.1825156  7.0719037  6.407488  6.856608  7.8716564  8.829419  
6.622335  6.277051  66.61289 ]
```

average error between angles

*Thank
You*