

3D Human Pose Estimation

Farhanur Rahim Ansari, Vidhey Oza,
Minji Lee





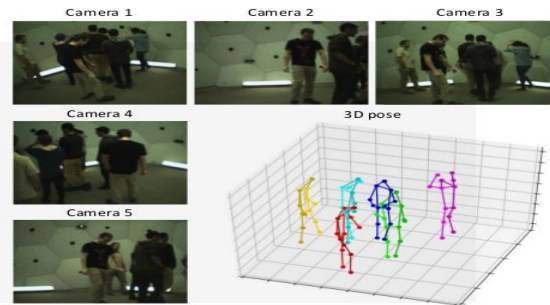
Project Goal

Perform 3D human pose estimation on monocular RGB images and videos, and make an interactive tool that helps in using this technology with convenience.

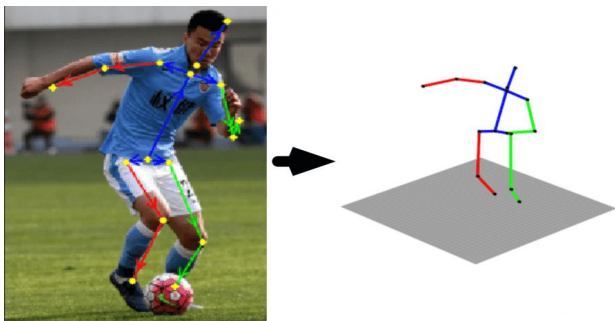
Where is this useful?



Robotics



Surveillance



Motion Capture



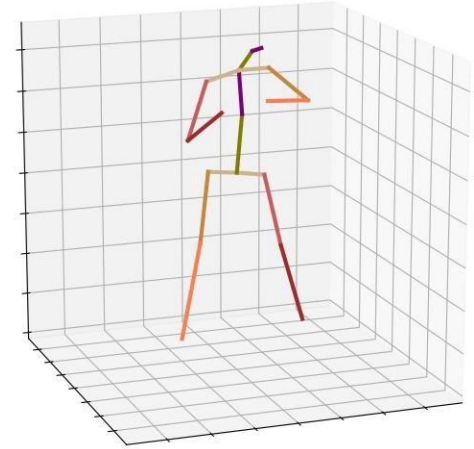
CGI

Challenges - Self-Occlusion + Depth Ambiguity

Cricket batsman practicing different shots

Self-occlusion- one of the arm and leg are hidden behind the batsman's body

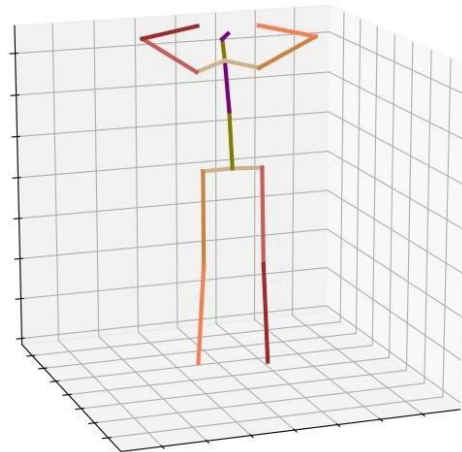
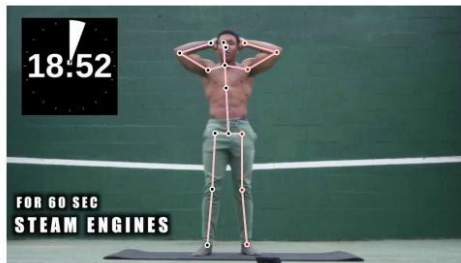
Depth Ambiguity- How far the different body parts are from the camera is difficult to access



Challenges - Jitter

Man doing jumping jacks and stem engines

Jitter- Because of the lesser number of frames to cover this complicated exercise the final output is a bit jittery or shaky



Approach

- Two-step method first predict 2D keypoints from RGB images and then lift them to 3D poses.

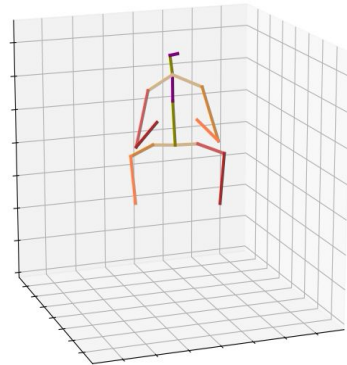
**Two-step
Method:**



2D Keypoint
Prediction



3D Pose
Reconstruction



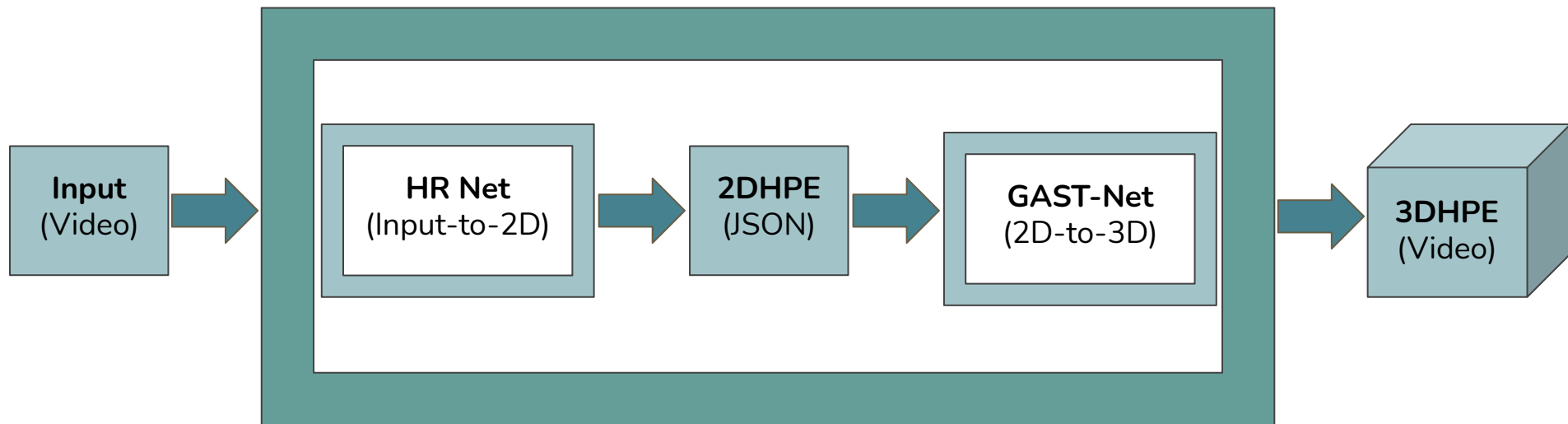


Why two-step methods?

- Compatible with existing 2D pose estimation methods
- Avoids influence of background and human surface features
- 2D pose acts as auxiliary output for better convergence
- Has better generalization in the wild



3D-HPE Pipeline





Dataset

HR-Net

1. COCO
 - 42.7 GB
 - 330K images
 - 250K people
2. MPII Dataset
 - 12.9 GB
 - 25K images
 - 40K people

GAST-Net

1. Human3.6M
 - 3.6m 3D human poses
 - 17 activities
 - 11 actors
2. HumanEva-I
 - 13.6 GB
 - 6 activities
 - 4 actors

HRNet

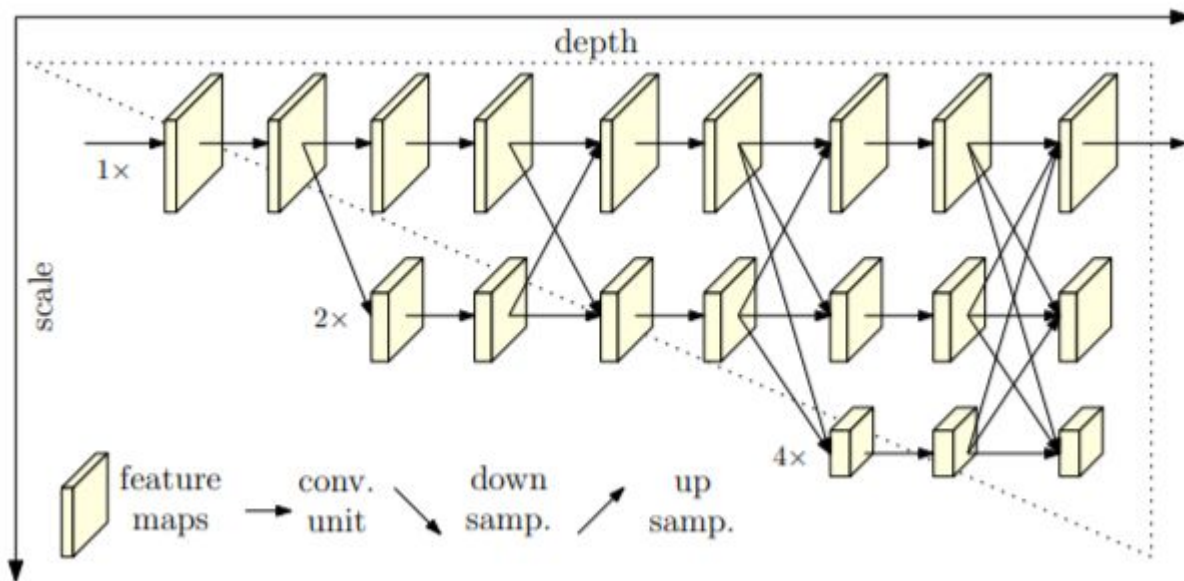


Fig: Architecture of HRNet Framework

GAST-Net

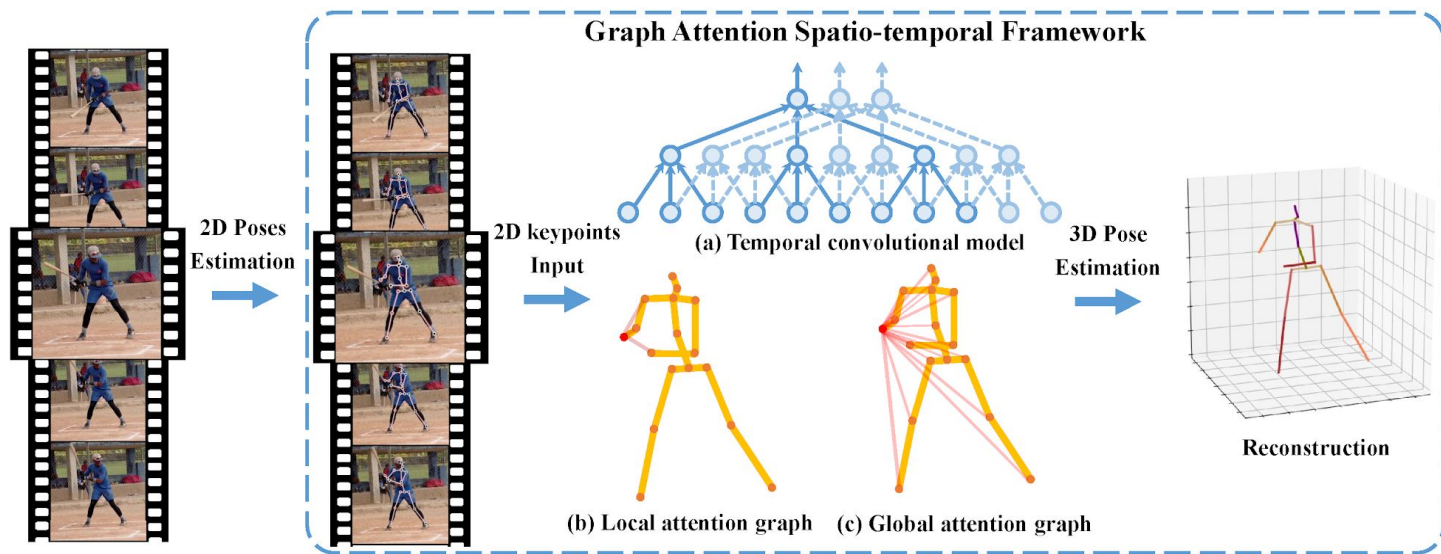


Fig: Schematic overview of GAST-Net Framework



Quantitative Performance

Evaluation Metrics-

- *mAP*: Mean Average Precision
- *MPJPE*: Mean Per Joint Positioning Error

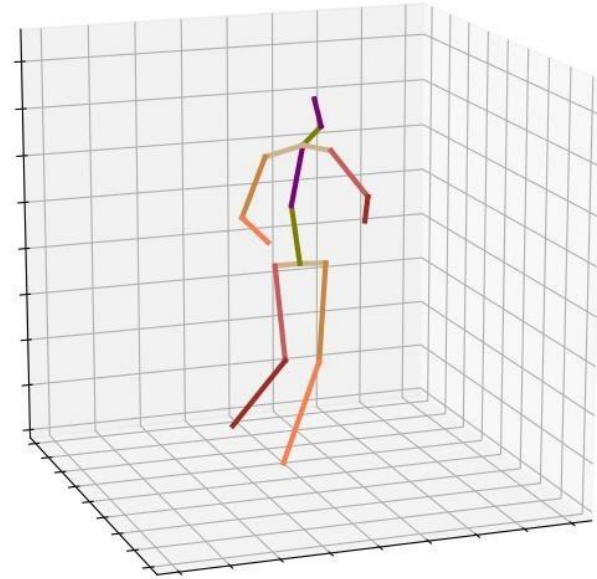
mAP	COCO	MPII
HRNet	91.5	92.3
CPN	74.9	77.0
Integral Pose	67.8	-
SimpleEnsemble	-	91.5

MPJPE	Human3.6M	HumanEva-I
GAST-Net	23.11	21.2
Pavlo	34.5	35.2

Observation: HRNet and GAST-Net both outperform the benchmarks (or are very close), but their real novelty is being able to achieve similar performance on random unseen data

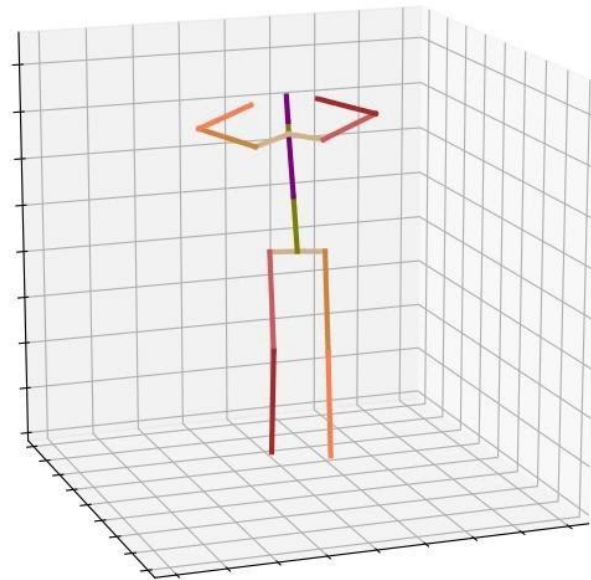
Issues Fixed - Occlusion

Self-occlusion occurring due to side profile view overcome

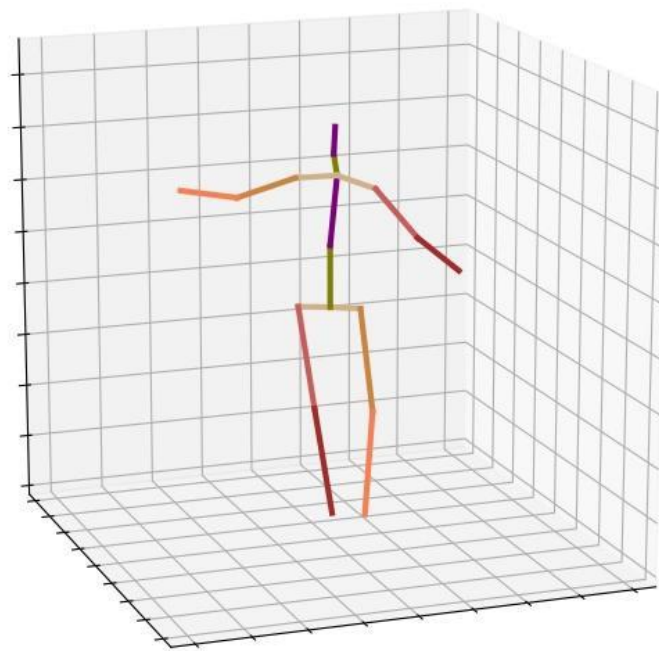


Issues Fixed - Jitter

Due to proper formatting in pipeline, jitter is reduced



How can I use this?





Building Web Applications



- Python-based free and open-source web framework
- Version 3.1.4
- Pre-built sqlite database
- Made index.html, after_index.html



- Ubuntu 18.04.4
- 15.3GB memory
- CPU



Demo Time!!

3D-HPE

3D Human Pose Estimation

Caption:

Video: Choose File No file chosen

Upload

Home Page



Conclusion

- Did a thorough research of the field
- Tried working on multiple datasets and models
- Set up pipelines for project (HR-Net, GAST-Net)
- Fixed issues (Jittery, Occlusion, and depth ambiguity)
- Set up web application with Django and AWS



Thank you!