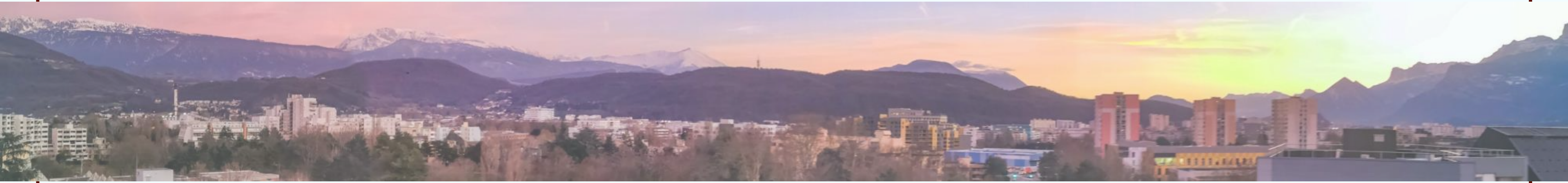
Design and evaluation of a biomechanically consistent method for markerless kinematic analysis of sports motion

N.B.: Introduction in French, the rest in English

Doctoral school: Mathematics, Information Sciences and Technologies, and Computer Science
Specialty: Computer sciences
University: Université Grenoble Alpes (UGA)
Laboratories: Laboratoire Jean Kuntzmann (Grenoble), Laboratoire Pprime (Poitiers)

Jury members:
Valérier PERRIER: Chairperson (LJK, Grenoble INP)
Steffi COLYER: Main examiner (University of Bath, England)
Richard KULPA: Main examiner (Université Rennes 2)
Mickaël BEGON: Examiner (Université de Montréal, Canada)
Bruno WATIER: Examiner (Université Toulouse III - Paul Sabatier)
François BRÉMOND: Examiner (INRIA Méditerranée)
Lionel REVERET: Supervisor (Inria Rhône-Alpes)
Mathieu DOMALAIN: Co-supervisor (Université de Poitiers)

David Pagnon PhD Defense, March 10th, 2023, Grenoble (France)



01

Introduction

What is MoCap?
Why do we need markerless in sports?

02

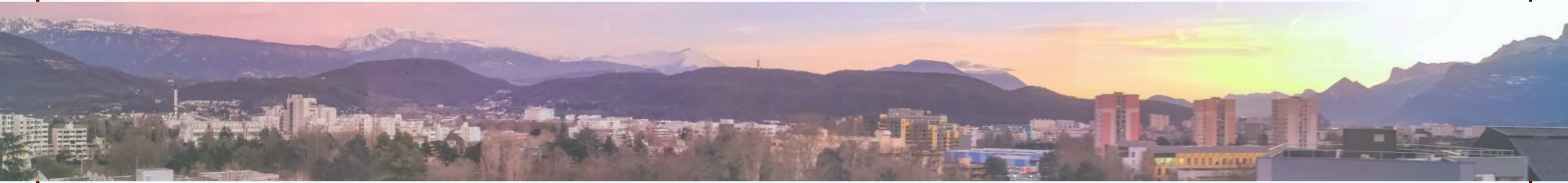
From 2D images to 3D joint angles

Crossover of numerous research fields.
Building an open-source solution.

03

Robustness

Does it actually work? Comparing results under challenging capture conditions



04

Accuracy

Comparison to a marker-based process.

05

Application case

Boxing, using unsynchronized and uncalibrated GoPros.

06

Perspectives

Enhancing the markerless process.
Data fusion when video is challenged

The Lord of the Rings, Cinema-news.com
Assassin's Creed, Ubisoft

Introduction

What is MoCap?

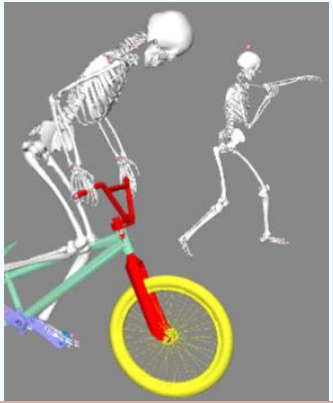
This is Motion Capture = MoCap

More precisely, marker-based MoCap
Markers are tracked to render motion


Assassin's creed, Ubisoft

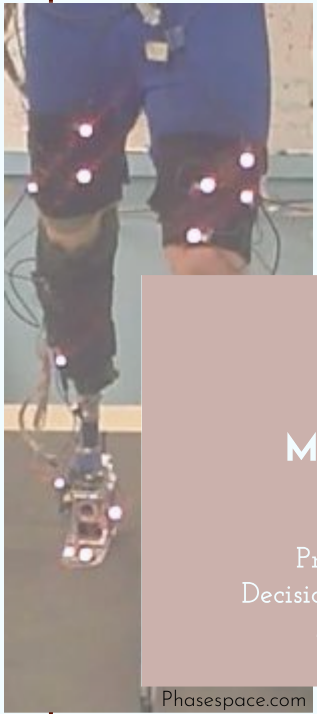
Film and game industries

Retargetting
the movements of an actor
to a character



Sports

Movement optimization,
Performance prediction,
Injury prevention

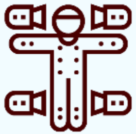

Phasespace.com

Medicine

Prosthetics,
Decision support for
surgery

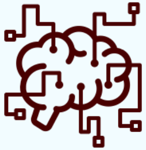
However,

marker-based systems are impractical in sports



Markers

- Cumbersome: hinders warm-up, movement and focus
- Prone to placement errors and to soft tissue artifacts
- Often occluded



System

- Sensitive to sunlight
- Complex setup and analysis
- Expensive

Marker-based analysis is:
- Used for research, not training
- Far from the reality of the field

Need for markerless analysis

Some attempts
of markerless MoCap

IMU sensors

Motion captured from accelerometer, gyroscope, and magnetometer. Ex: Xsens
☑ Easy to use, robust, cheap
✗ Drift, still something to wear

Remains an interesting compromise


Ubergizmo.com

RGB-D cameras

Red-green-blue + depth sensor. Ex: Kinect
☑ True markerless, easy to use, cheap
✗ Low framerate, short range, sensitive to sunlight

Not the first choice for biomechanics

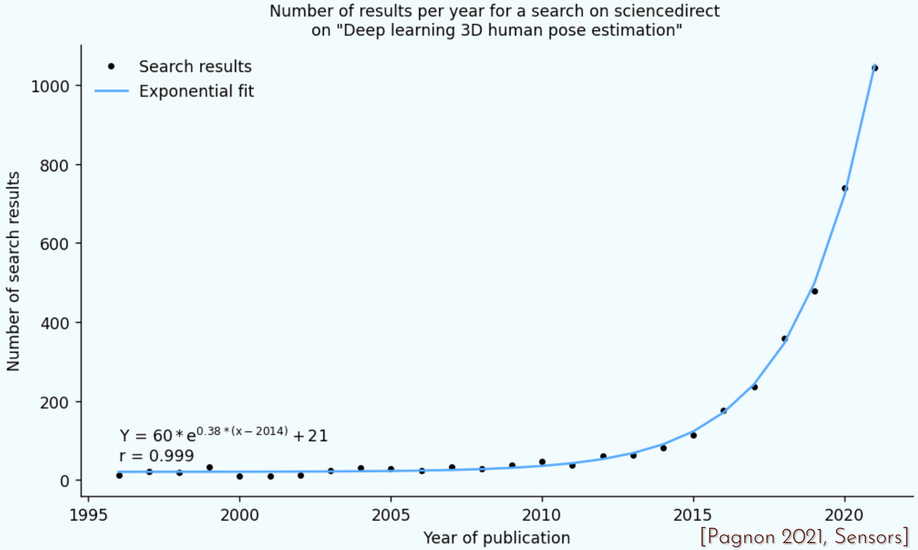

Dsdambuster.com

RGB camera network

First attempts by background subtraction + visual hull
✗ Needs synchronized, calibrated cameras
✗ Constraints on outfit and background colors
Even less convenient than marker-based systems?

But then came

deep-learning pose estimation on RGB cameras



[Pagnon 2021, Sensors]

[Cao 2019, IEEE]

- ☑ No marker placement, no soft tissue artifacts
- ☑ No special outfit
- ☑ Extremely robust to capture conditions
- ☑ Works with cheap cameras

Deep learning-based markerless analysis?

Current state of the art

of markerless biomechanics

1

2D pose estimation

The foundation

- OpenPose: [Cao 2019]
  The most widespread one. Extremely robust, reasonably accurate and fast.
  Bottom-up method: ↗ number of people ⇏ ↗ computing time.

Top-down: AlphaPose [Fang 2017]

Bottom up: OpenPose [Cao 2019]

- BlazePose (Mediapipe) [Bazarevsky 2020]

- DeepLabCut [Mathis 2018]

+ YOLO-pose [Wang 2022], SLEAP [Pereira 2022], ...

2

Monocular 3D pose estimation

3D: exists!

- BlazePose
- SMPL suite

3

Multi-view 3D pose estimation

Occlusions: exist!

- FreeMocap
- EasyMocap
- AniPose

4

Multi-view 3D joint angles

Skeleton: exists!

- Theia
- Pose2Sim
- OpenCap

Current state of the art

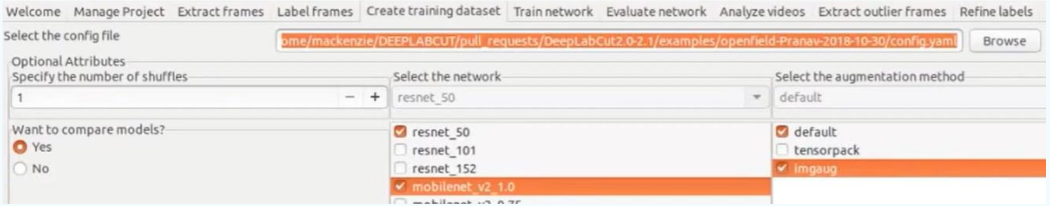
of markerless biomechanics

1

2D pose estimation

The foundation

- **OpenPose:** [Cao 2019]
- **BlazePose (Mediapipe)** [Bazarevsky 2020]
  Extremely fast but less robust and accurate, and only single person detection.
  Detects a person on a frame, then tracks them until they are lost.
- **DeepLabCut:** [Mathis 2018]
  Train model with your own keypoints.
  Choose architecture and parameters of the neural network.



+ YOLO-pose [Wang 2022], SLEAP [Pereira 2022], ...

2

Monocular 3D pose estimation

3D: exists!

- BlazePose
- SMPL suite

3

Multi-view 3D pose estimation

Occlusions: exist!

- FreeMocap
- EasyMocap
- AniPose

4

Multi-view 3D joint angles

Skeleton: exists!

- Theia
- Pose2Sim
- OpenCap

Current state of the art

of markerless biomechanics

2

Monocular 3D pose estimation

Sports movements are 3 dimensional!

- **BlazePose** (MediaPipe): Also detects 3D keypoints from single camera
  Probabilistic guess: mostly interesting when accuracy is not critical.

Mediapipe

- **SMPLify suite**: 2D keypoints $\xrightarrow{\text{Regressor}}$ 3D SMPL shape [Bogo 2016]
  Common issue: person leaning forward
  Newer regressor: STRAPS [Sengupta 2020]; newer shape model: STAR [Osman 2020].

[Rempe 2020]

1

2D pose estimation

The foundation

- OpenPose
- BlazePose
- DeepLabCut

3

Multi-view 3D pose estimation

Occlusions: exist!

- FreeMocap
- EasyMocap
- AniPose

4

Multi-view 3D joint angles

Skeleton: exists!

- Theia
- Pose2Sim
- OpenCap

Current state of the art

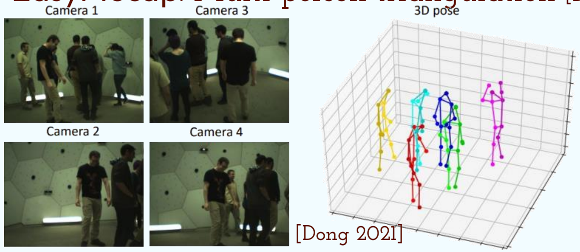
of markerless biomechanics

3

Multiview 3D pose estimation

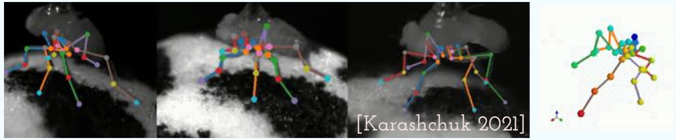
Several viewpoints are needed to solve occlusions accurately!

- **FreeMocap**: Naive but user-friendly OpenPose triangulation [Matthis 2022]
- **EasyMocap**: Multi-person triangulation [Dong 2021]



[Dong 2021]

- **AniPose**: Triangulates data from DeepLabCut results [Karashchuk 2021]



[Karashchuk 2021]

1

2D pose estimation

The foundation

- OpenPose
- BlazePose
- DeepLabCut

2

Monocular 3D pose estimation

3D: exists!

- BlazePose
- SMPL suite

4

Multi-view 3D joint angles

Skeleton: exists!

- Theia
- Pose2Sim
- OpenCap

Current state of the art

of markerless biomechanics

4

3D

Multiview 3D joint angles

Constraints on segments and joints need to be satisfied!

- **Theia**: [Kanko 2021]
  First on the market, user-friendly, rather accurate.
  Not open-source! Black box, not flexible.
- **Pose2Sim**: [Pagnon 2021]
  Our solution. Robust, accurate, and flexible.
- **OpenCap**: [Uhlrich 2022]
  Estimates coordinates of 43 markers from triangulated OpenPose keypoints.
  Less cameras needed, but model might not work as well on untrained tasks.
  User friendly, but requires sending data to their servers.

Commercial but not peer-reviewed: The Captury, Simi Motion.
Peer-reviewed but not released: [Needham 2022]

1

2D

2D pose estimation

The foundation

- OpenPose
- BlazePose
- DeepLabCut

2

3D

Monocular 3D pose estimation

3D: exists!

- BlazePose
- SMPL suite

3

3D

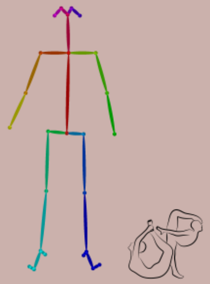
Multi-view 3D pose estimation

Occlusions: exist!

- FreeMocap
- EasyMocap
- AniPose

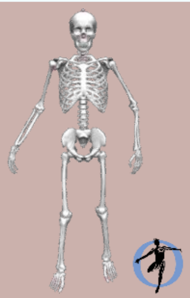
Objectives

Bridging the gap between
Deep learning and Biomechanics



OpenPose
Deep learning
Video → 2D pose
Inaccurate?

Pose2Sim
Computer vision



OpenSim
Biomechanics
Markers → 3D joint angles
Incompatible with sports?

Objectives

Approaching an
ideal motion capture system

Ideal motion capture system: [Atha 1984, Appl Ergon]

? Accurate
? No interference with natural movement
? Simple
? Quick set up and analysis
? Cheap

MoCap performance characteristics: [Moeslund 2001, CVIU]

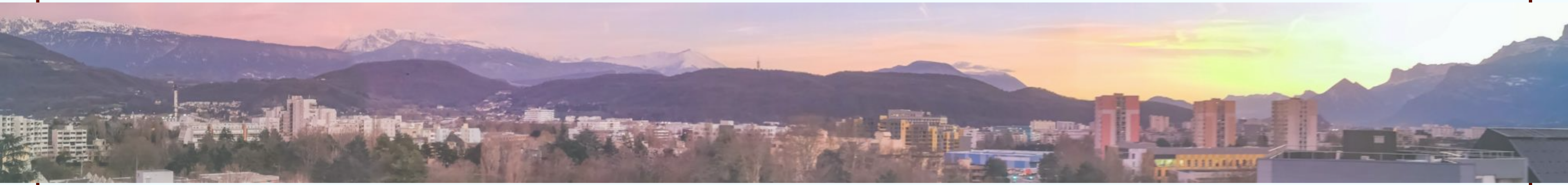
? Robustness
? Accuracy
? Speed

- Developing and publishing an open-source tool
- Challenging its robustness
- Assessing its accuracy
- Testing it in sports settings

01
Introduction
What is MoCap?
Why do we need markerless in sports?

02
From 2D images to 3D joint angles
Crossover of numerous research fields.
Building an open-source solution.

03
Robustness
Does it actually work? Comparing results under challenging capture conditions



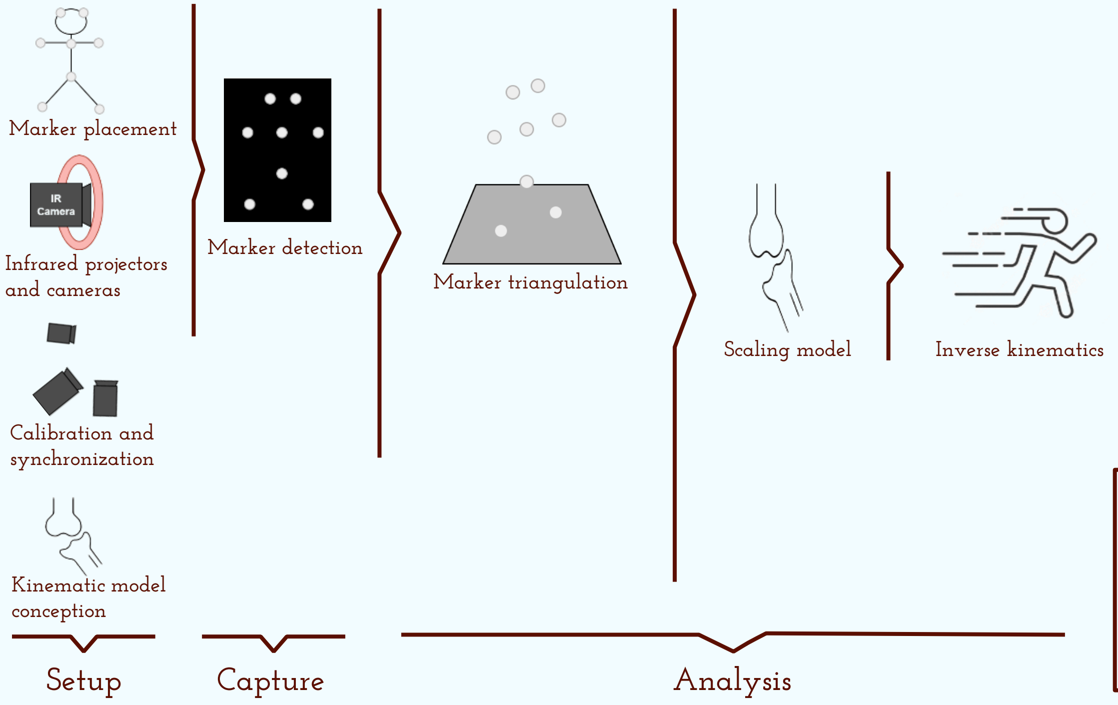
04
Accuracy
Comparison to a marker-based process.

05
Application case
Boxing, using unsynchronized and uncalibrated GoPros.

06
Perspectives
Enhancing the markerless process.
Data fusion when video is challenged

Preamble

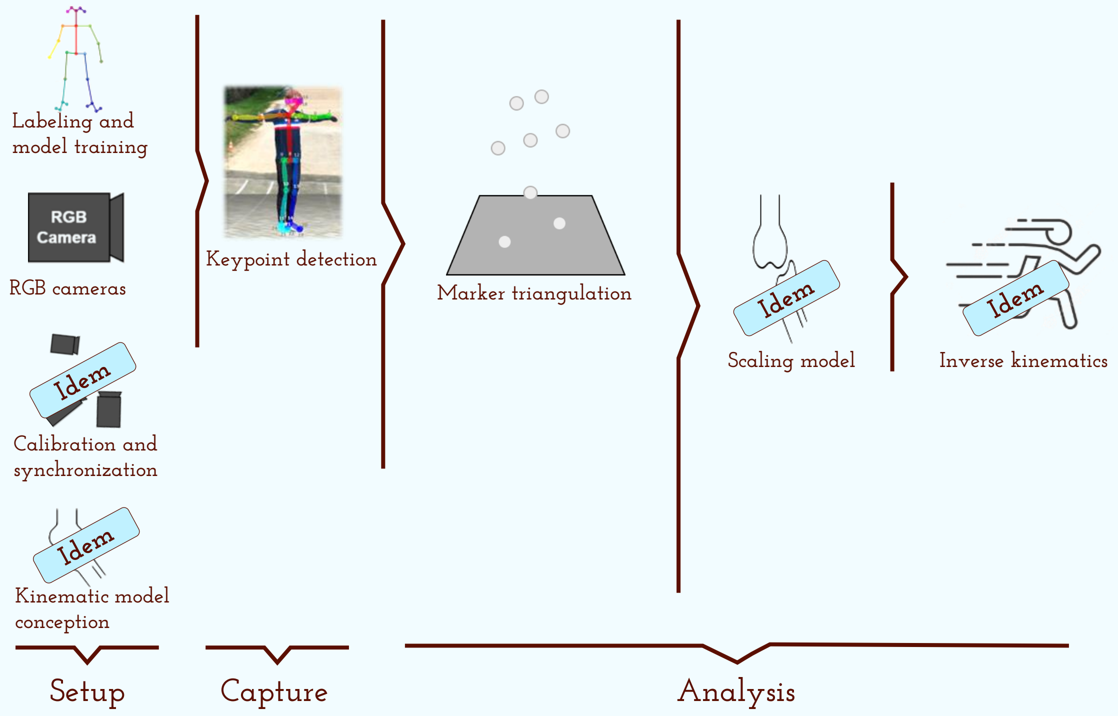
How does marker-based MoCap work?



Definitions:
Calibration: Camera properties and placement
Kinematics: Motion analysis
Direct kinematics: Marker positions from joint angles
Inverse kinematics: Joint angles from marker positions

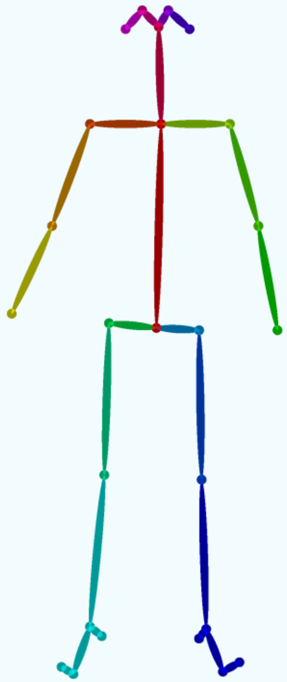
A markerless implementation

[Pagnon, 2022, JOSS] Pose2Sim, an open-source package



2D pose estimation

With deep learning



Labelling — Keypoints labelled by humans on 1000s of images

Training — After choosing a network architecture and its parameters:
- Network guesses keypoint positions
- Error calculation → update weights
- Several iterations on different images until correct positions

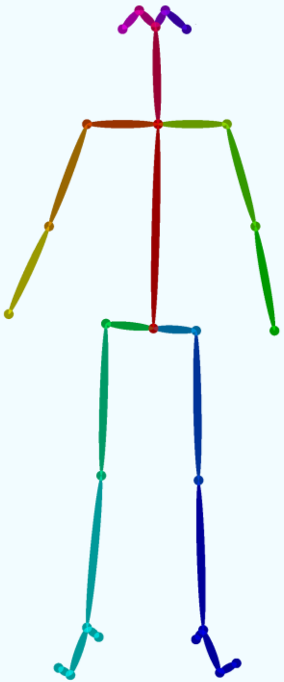
➜ The network has learned: model ready to be used

Inferring — Detecting keypoints on real data

\+ Hand-labelled right shoulder
● First guesses during training

2D pose estimation

With deep learning



Labelling **Training** **Inferring**

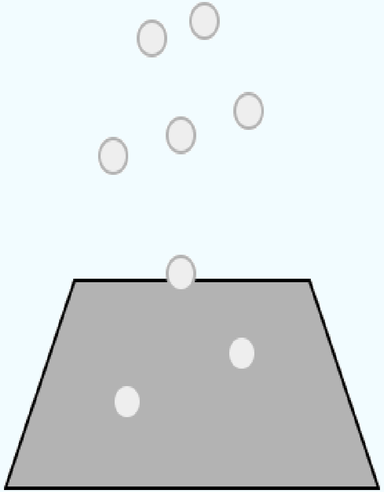
Several options
- **Whole process from scratch:** Tedious, unnecessary
- **Transfer learning:** Label less images, train less long (e.g. DeepLabCut)
- **Pretrained model:** Use as is (e.g. OpenPose)

Pose2Sim validation used the most accurate model from OpenPose (body 25B)
Other OpenPose models, BlazePose, DeepLabCut, and AlphaPose also supported.

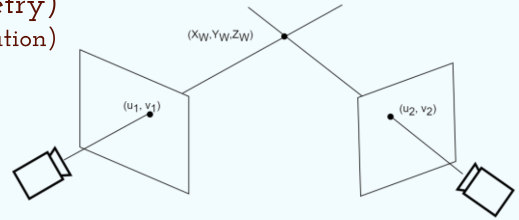
\+ Hand-labelled right shoulder
• First guesses during training

Triangulation

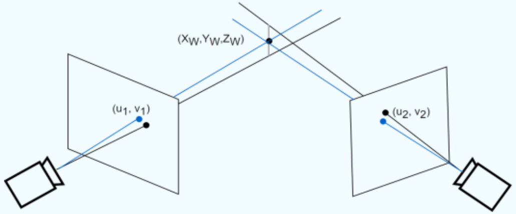
With computer vision



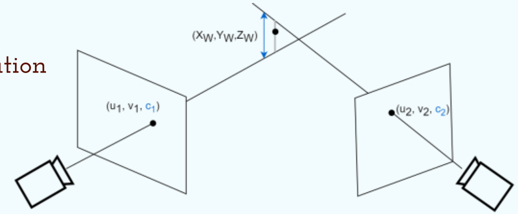
- **Ideal case:** Epipolar geometry (simple trigonometry)
  When rays intersect (if perfect cameras & perfect 2D estimation)



- **In real life:** e.g. Direct linear Transform (DLT)
  Rays never perfectly intersect: minimize reprojection error

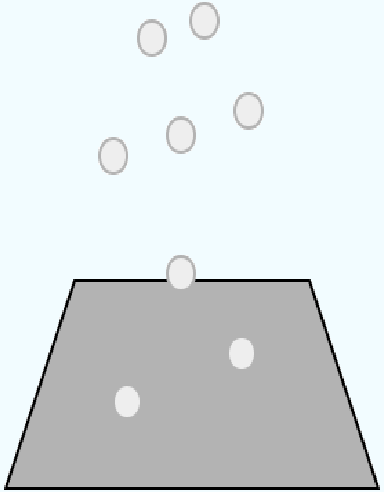


- **Proposed method:** Weighted DLT
  With confidence score that comes with each keypoint estimation
  More accurate, fast



Triangulation

With computer vision

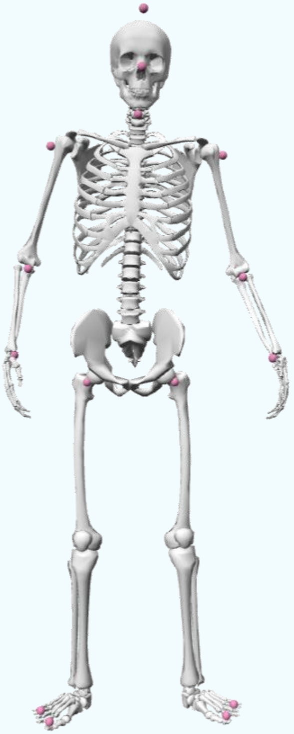


Pose2Sim:

- Triangulates with **weighted DLT**
- Automatically chooses **the right person** to triangulate
- Is robust to **person exiting** and entering camera field
- Offers **flexible settings** for triangulation and filtering
- Gives **precise feedback**: which keypoints, cameras, and frames pose problem?

Kinematic optimization

With biomechanics



Model

Triangulated keypoints positions are **approximate:**

- Systematic offset between keypoints and joint centers [Needham 2021]
  → Pose2Sim: Offset taken into account
- Keypoint location on body may vary from frame to frame
  → Pose2Sim: Bone lengths are fixed

Triangulated keypoints are **sparse**:

21 keypoints, 20 joints × 3 (or 6) degrees of freedom: need for constraints

- Pose2Sim: Knee abduction & internal rotation constrained to flexion [Rajagopal 2015, IEEE]
- Pose2Sim: Each vertebra constrained to the previous one [Beaucage-Gauvreau 2019, Comp Met]
- Pose2Sim: Angle limits are defined for each joint

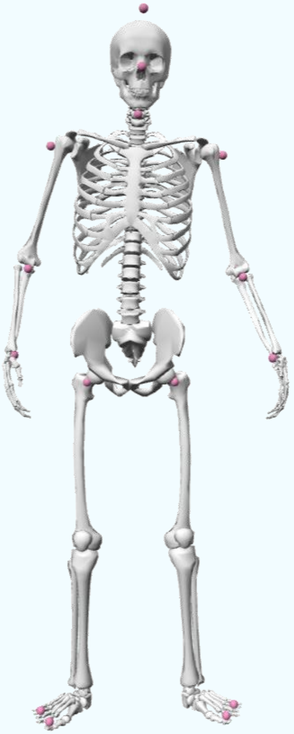
Scaling

Inverse kinematics

[Pagnon 2022, Sensors]

Kinematic optimization

With biomechanics



Model

Triangulated keypoints positions are **approximate and sparse**

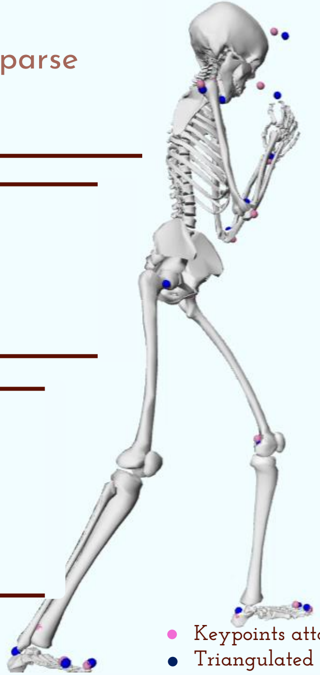
Scaling

Everyone has different size and proportions:
- Distances between markers determine segment dimensions
- Keypoint positions are subject- and operator-agnostic
  → Pose2Sim: No need for adjustments once the model is defined

Inverse kinematics

Scaled model follows keypoint movements
Joint angles optimized to minimize distance between triangulated keypoints and model keypoints

$$min \sum_{k \in keypoints} \left\| Coord^{trig} - Coord^{model}(angles) \right\|^2$$



- Keypoints attached to the model
- Triangulated keypoints to follow

Pose2Sim summary

Crossover of several research fields:
- 2D pose estimation with **deep learning**
- Robust triangulation with **computer vision**
- Consistent kinematics with **biomechanics**

Pose2Sim is:
- Open-source, with extensive documentation
- Flexible:
  - any 2D pose estimation,
  - custom triangulation parameters,
  - custom kinematic model
- Gives **feedback** to user

Build on Win-MacOS-Ubuntu with Python 3.7-3.10 passing | pypi package 0.3.4
downloads 12k | stars 50 | License BSD 3-Clause | issues 3 open | issues 17 closed
JOSS 10.21105/joss.04362 | DOI 10.5281/zenodo.7606664

- **Continuously integrated:** automatic tests at each commit
- **Easy to install:** pip install pose2sim
- **Used** : 13k downloads, 51☆, 20 issues
- **Peer-reviewed** in JOSS [Pagnon 2022b, JOSS]

01
Introduction

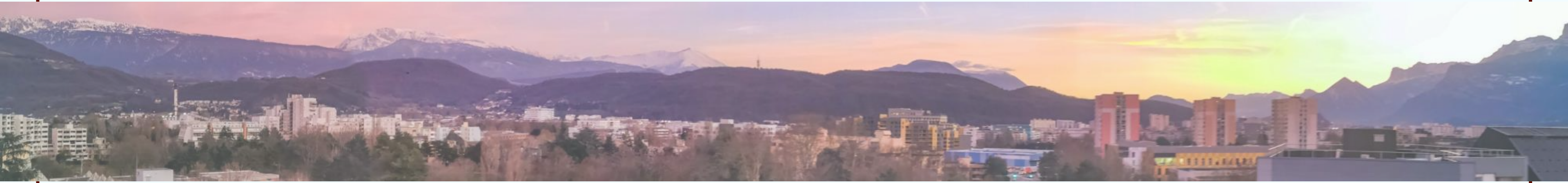
What is MoCap?
Why do we need markerless in sports?

02
From 2D images to 3D joint angles

Crossover of numerous research fields.
Building an open-source solution.

03
Robustness

Does it actually work? Comparing results under challenging capture conditions



04
Accuracy

Comparison to a marker-based process.

05
Application case

Boxing, using unsynchronized and uncalibrated GoPros.

06
Perspectives

Enhancing the markerless process.
Data fusion when video is challenged

Exploring robustness

Changing capture conditions?
- Involves numerous capture sessions
- Involves assuming no subject variability

Unless… Conditions were changed **after** capture

Kinovis platform:
68 video cameras (+ 20 marker-based ones)

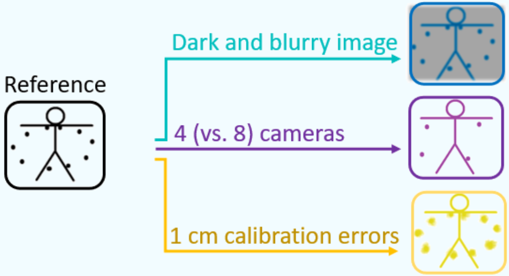
1. Capture of body shape and texture
2. Scene recreation **with different conditions**
3. Pose2Sim workflow

Comparison of kinematics
under several challenging conditions

Robustness assessment

Robustness regarding:
- Activities: walking, running, cycling
- Subject exiting the field of view
- Additional persons entering the scene

Assessed separately against:
- Image quality → dark scene, defocused objectives
- Occlusions → filming with half as many cameras
- Poor calibration → random 1 cm residuals



[Pagnon 2021, Sensors]

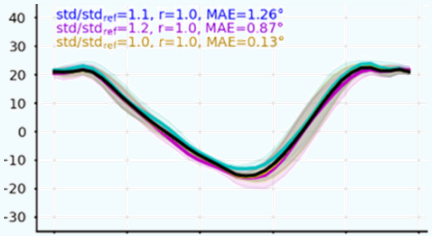
Robustness results

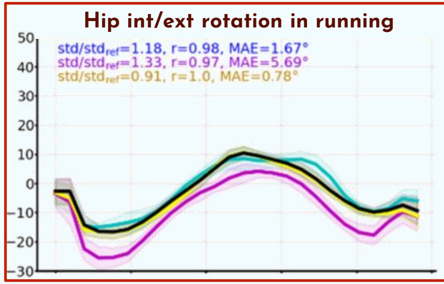
Very good agreement with reference condition

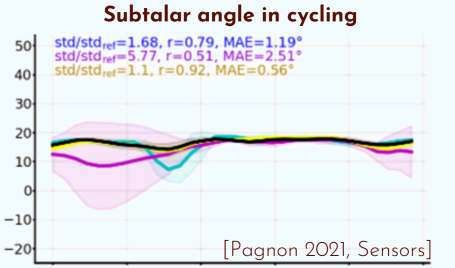
- Across all simulated conditions, activities, planes of movement, and joints:
  MAE < 5.7°

- 1 cm calibration error → virtually no difference:
  r=1.0, MAE<1°, std<2.5°

- Fewer cameras and poor image quality → more challenging, especially out of the flex/ext plane, especially on the cycling task







[Pagnon 2021, Sensors]

⚠ 1 single subject, 10 cycles per task
⚠ Simple sagittal movements
⚠ Virtual scene

Definitions:
MAE: Mean absolute error (re. reference)
r: Pearson's correlation coefficient (with reference condition)
std: Standard deviation

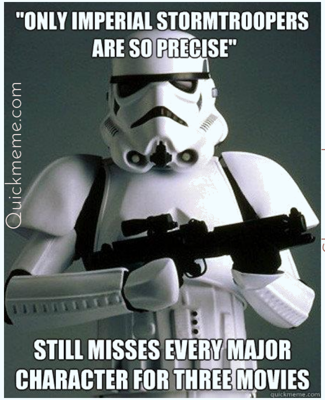
Robustness summary

Pose2Sim is robust to:

- Calibration errors: 1 cm makes no difference, while < 1 mm advised with markers
- Image quality: only mildly sensitive to lighting and focus, while marker-based systems hardly work in non-optimal conditions (e.g., outdoors)
- Occlusions: makes guess, while markers are either visible or not

[Pagnon 2021, Sensors]

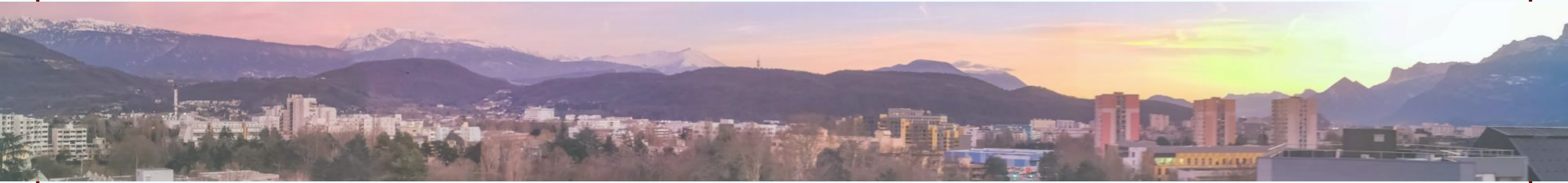
Results are similar regardless of the condition.
Are they accurate?



01
Introduction
What is MoCap?
Why do we need markerless in sports?

02
From 2D images to 3D joint angles
Crossover of numerous research fields.
Building an open-source solution.

03
Robustness
Does it actually work? Comparing results under challenging capture conditions



04
Accuracy
Comparison to a marker-based process.

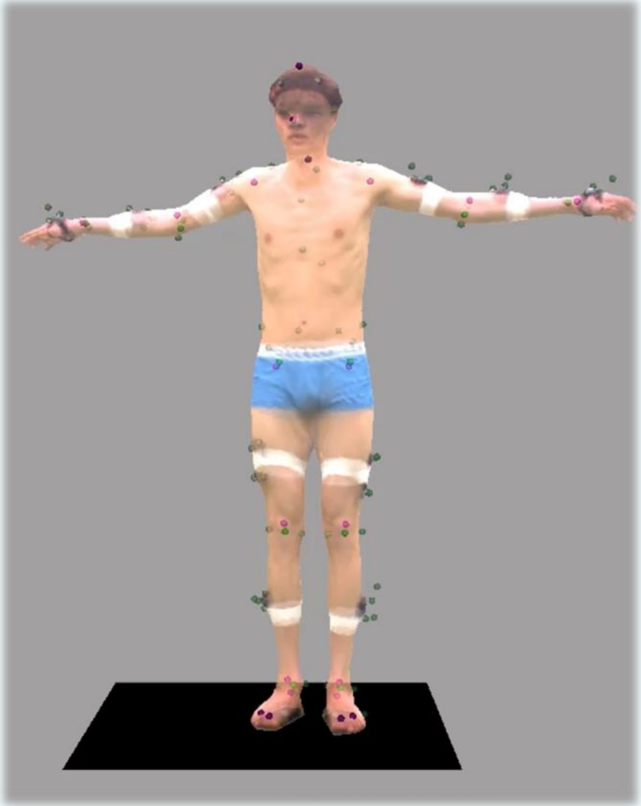
05
Application case
Boxing, using unsynchronized and uncalibrated GoPros.

06
Perspectives
Enhancing the markerless process.
Data fusion when video is challenged

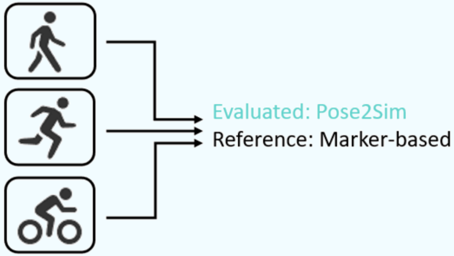
Accuracy assessment

Comparison with a marker-based method

Systematic position bias up to 5 cm (knee, hip). Reported by [Needham 2021]
→ systematic angle bias?
Could be precise, but inaccurate...



Marker-based protocol:
20 marker-based cameras (vs. 8 virtual video cameras)
83 markers + 8 calculated joint centers (vs. 21 keypoints)
Same kinematic model
Marker adjustments during scaling (vs. subject agnostic & no soft tissue artifacts)

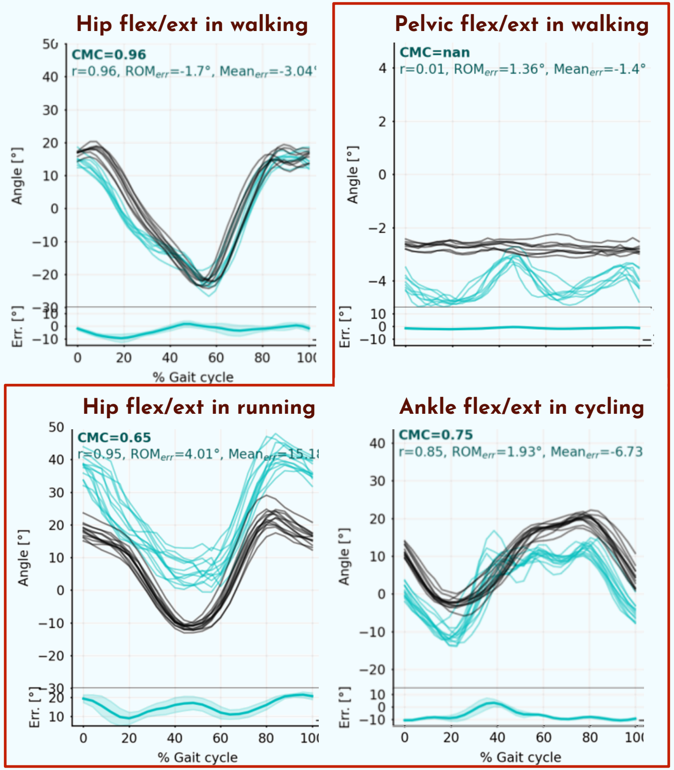


Accuracy results

Grand mean errors:
- 3.0° in walking
- 4.1° in running
- 4.1° in cycling

Very good agreement in flex/ext
- except for pelvis
- except for offsets in hip in running (15°) and ankle in cycling (6.7°)

Less good on other planes of movement, esp. in cycling.



Comparison

with other systems

Walking	RMSE (range)
Theia [Kanko 2021]	6.4° (3.3°-11°)
Pose2Sim [Pagnon 2021]	4.9° (3.1°-6.6°)
[Needham 2022]	4.9° (2.9°-6.0°)
OpenCap [Uhlrich 2022]	4.9° (2.9°-6.0°)

⚠ Highly constrained kinematic model
Assumes similar joint mechanics within all human beings

⚠ Marker-based methods ≠ gold-standard
Unlike biplanar videography / MRI / bone-anchored pins / 3D ultrasound
9° errors vs gold standard [Kessler 2019]
10° inter-operator errors [Gorton2009]
3° soft tissue artifacts [Benoit2015]
3° joint position errors [Leboeuf2019]

Definition:
RMSE: Root mean square error (re. reference)

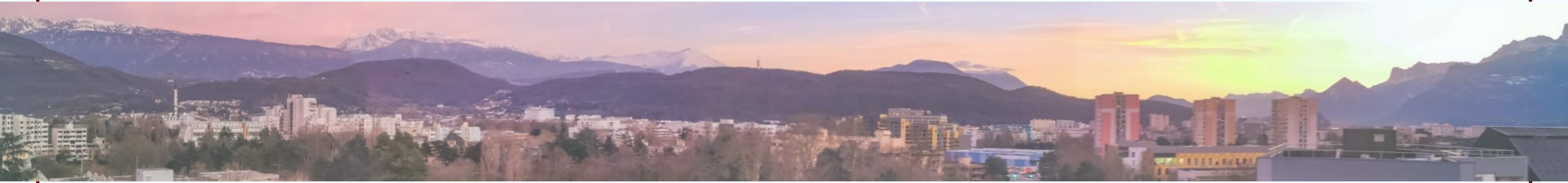
Good accuracy, good robustness. [Pagnon 2022a, Sensors]
But pelvic region less accurate, while sometimes investigated in gait analysis [Saunders 1953]

Lower-body analysis of simple motion:
What about real sports movements, in a real setting?

01
Introduction
What is MoCap?
Why do we need markerless in sports?

02
From 2D images to 3D joint angles
Crossover of numerous research fields.
Building an open-source solution.

03
Robustness
Does it actually work? Comparing results under challenging capture conditions



04
Accuracy
Comparison to a marker-based process.

05
Application case
Boxing, using unsynchronized and uncalibrated cheap cameras.

06
Perspectives
Enhancing the markerless process.
Data fusion when video is challenged

Addressing sports

Sports movements:

3D, fast, full-body

⟶ Capture shadow-boxing sequence: 6 repetitions by 3 elite athletes

⟶ Determine and measure Key Performance Indicators (KPIs)

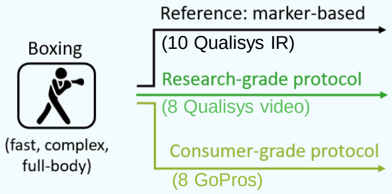
Sports conditions:

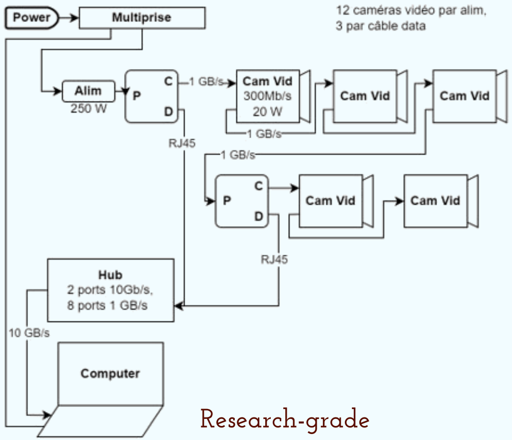
Need for fast, not cumbersome, cheap setup

⟶ GoPros: lightweight, wireless, consumer-grade cameras

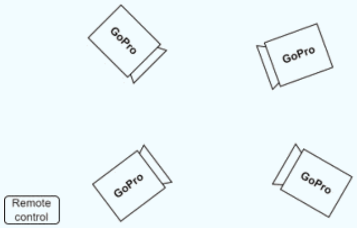
⟶ But uncalibrated & unsynchronized

Comparison study:





Research-grade

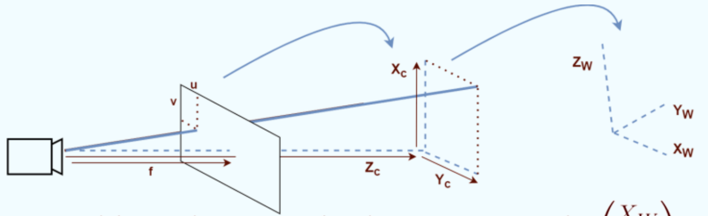


Consumer-grade

1. Are KPIs accurately retrieved by markerless analysis?
2. Is consumer-grade hardware good enough?

Calibration

Matching Image / World coordinates



$$Z_C \times \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_u & 0 & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R}_{3\times 3} & \mathbf{T}_{3\times 1} \end{pmatrix} \begin{pmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{pmatrix}$$

Image 2D coordinates — Intrinsic parameters (+distortion) — Extrinsic parameters — **World 3D coordinates**

Intrinsic parameters: Image 2D → Camera 3D
What kind of camera is it?
Can be done off-line

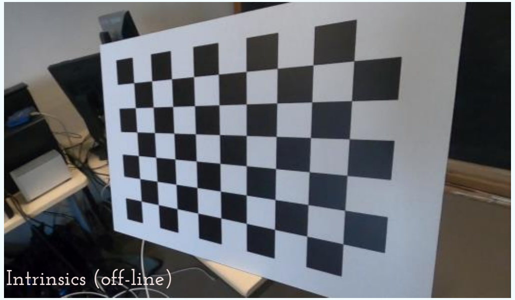
Extrinsic parameters: Camera 3D → World 3D
Where is the camera?
Requires on-field information

Calibration

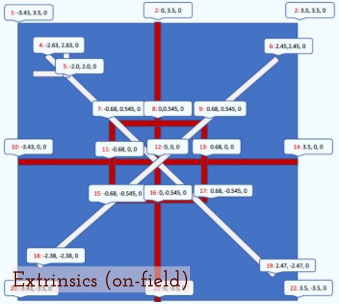
Matching Image / World coordinates

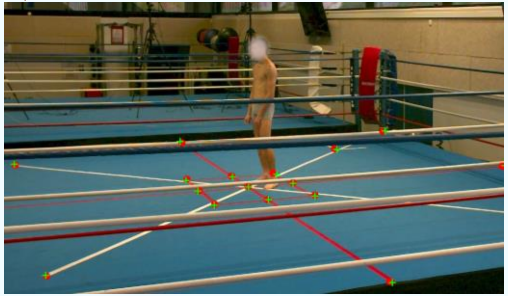
Calibration = main stumbling block for users
→ Need for user-friendly tool

1. **Intrinsic parameters**: with checkerboard


Intrinsics (off-line)

2. **Extrinsic parameters**: with intrinsic parameters
+ object of known dimensions on scene


Extrinsics (on-field)



3. **Check validity**: Compare clicked and reprojected points

Calibration errors: 2.5 cm

Not fully integrated in Pose2Sim yet (but workable scripts are provided)

Synchronization

With correlation of keypoint speeds

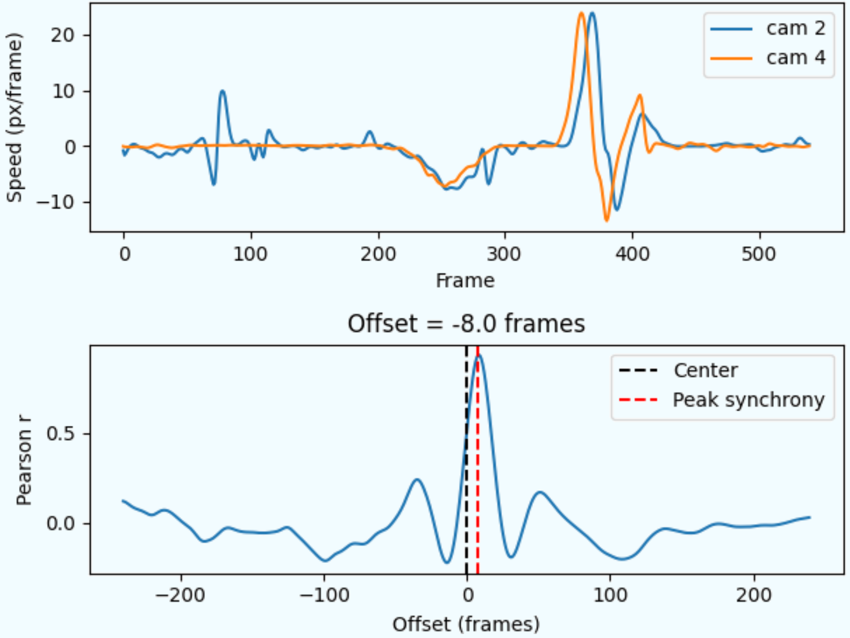
How to synchronize cameras?

- Cables: too cumbersome
- Flash or clap: not workable in competition
- Wireless or sound: not robust

→ Only use image information?

Assumption:
Two cameras are synchronized when vertical keypoint speeds are maximally correlated

Independently proposed by OpenCap [Uhlrich 2022]



User can choose one or several keypoints, and weigh them.
Tested with different kinds of movements
Offset applied to each view accordingly → Synchronization to the frame

Not fully integrated in Pose2Sim yet (but workable script is provided)

KPI accuracy results

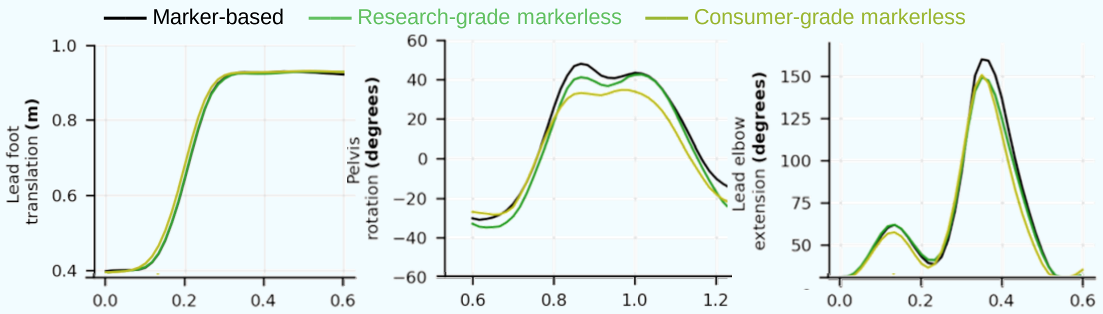
Research-grade markerless vs. marker-based:

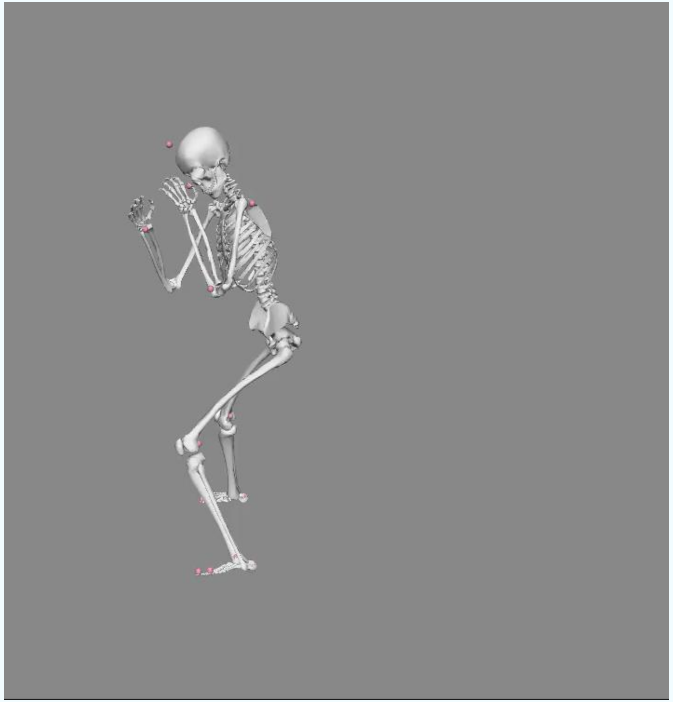
- **Onset times**: sub-frame agreement Errors <15 ms = 0.9 frames (@60fps)
- **Peak translations**: Sub-centimeter agreement Position errors: <1 cm and speeds: < -0,4m/s
- **Peak rotations**: Slight offsets, especially in pelvis & upper-body (up to 5.4°)

Consumer-grade hardware vs. research-grade:

Very slight differences, despite post-calibrated & post-synchronized

Rotations: up to 8.6° errors vs. up to 5.4°



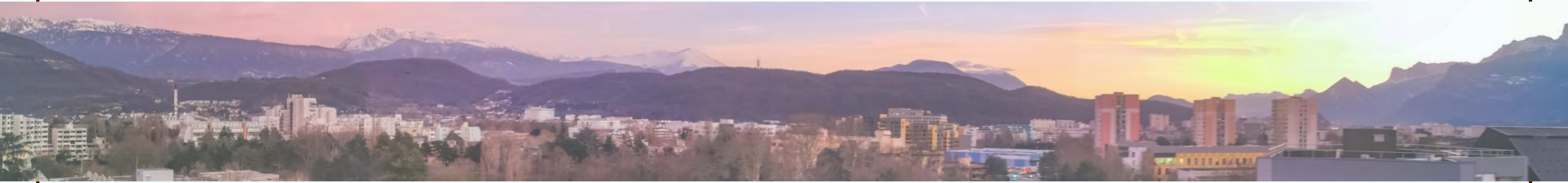


Application case summary

Markerless is relevant [Pagnon 2022c]
for whole-body, 3D, on-field sports analysis

Markerless with consumer-grade hardware is also relevant
- With lightweight, wireless, cheap cameras
- With relatively straightforward calibration and synchronization procedures

⚠ **Dearth of keypoints, systematic offset**: inaccuracies in pelvis, knee, elbow
⚠ **Constrained model in marker-based and markerless procedures**: inaccuracies in shoulder
⚠ **Single person analysis**: not workable in actual fights



Perspectives

How close are we from an ideal MoCap system?

Ideal motion capture system? [Atha 1984]

- Accurate → Need for more and better labelled keypoints
- No interference with natural movement → OK
- Simple → Need for user-friendly calibration and synchronization + GUI
- Quick set up → OK and analysis → Bottleneck: 2D pose
- Cheap → 5,000 € vs. 50,000 €

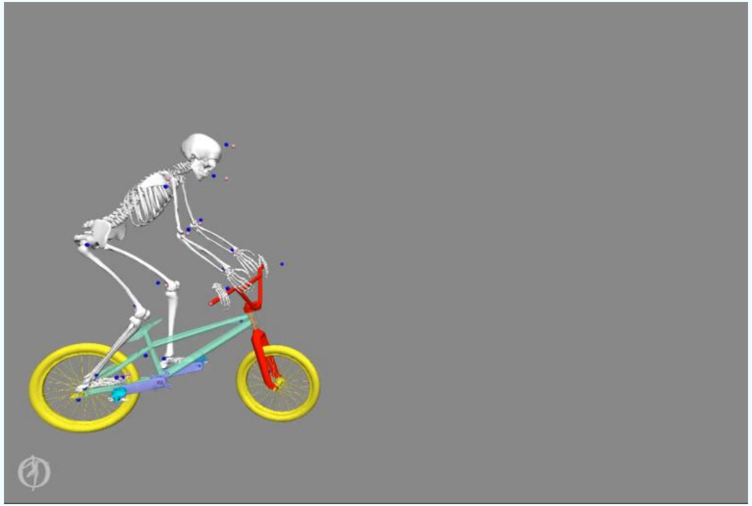
Perspectives

Going further in challenging cases

Example case: BMX race
- Vast field of view
- Large occlusions
- Equipment to detect

Preliminary study:
Pilot detection with OpenPose
Bike detection by DeepLabCut
} Pose2Sim → Unsatisfying results

Perspectives:
- Adding kinematic constraints between feet-pedals, hand-handle
- Positioning a few IMUs on feet and hands for orientation priors



01
Introduction

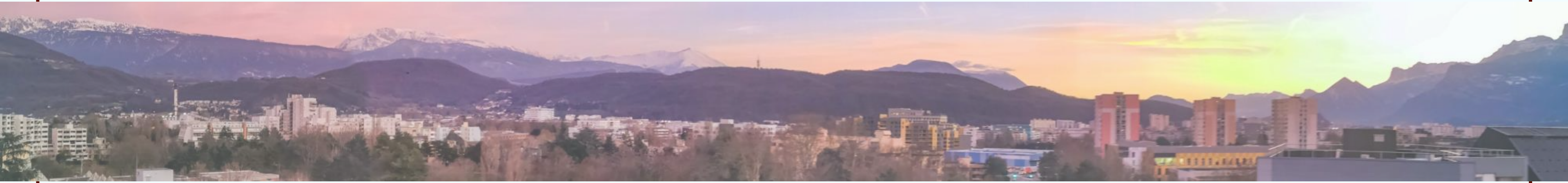
What is MoCap?
Why do we need markerless in sports?

02
From 2D pose to 3D joint angles

Crossover of numerous research fields.
Building an open-source solution.

03
Robustness

Does it actually work? Comparing results under challenging capture conditions



04
Accuracy

Comparison to a marker-based process.

05
Application case

Boxing, using unsynchronized and uncalibrated GoPros.

06
Perspectives

Enhancing the markerless process
Data fusion when video is challenged

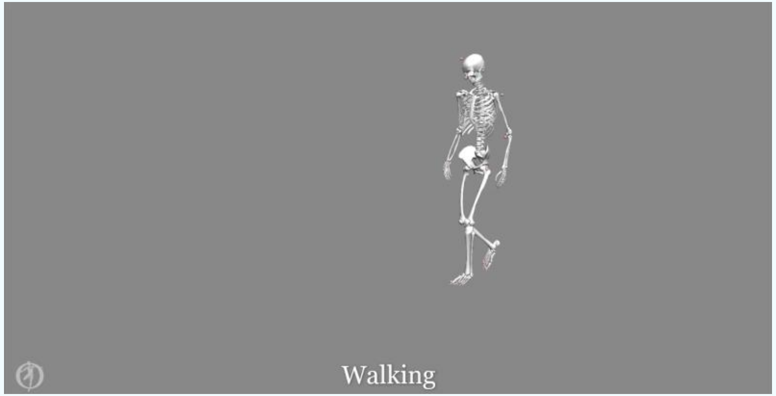
Conclusion

I am grateful:

- For this opportunity of mixing sports and science together (the cat is dead and alive in the same time!)
- That results ended up being satisfactory after a year and a half of unconclusive wandering
- For the opportunities of publishing my work Peer-reviewed articles: 3(+1), Conferences: 2
- And of releasing a workable tool Used by several labs and companies

Still so much more to look into!

Svgforest.com



Thank you

contact@david-pagnon.com
+33 6 19 86 06 57
http://david-pagnon.com

Bibliography

ATHA (1984) J *Current techniques for measuring motion.* Applied ergonomics
BEAUCAGE-GAUVREAU (2019) *Validation of an OpenSim full-body model with detailed lumbar spine for estimating lower lumbar spine loads during symmetric and asymmetric lifting tasks.* Computer Methods in Biomechanics and Biomedical Engineering
BENOIT (2015) *Surface marker cluster translation, rotation, scaling and deformation: Their contribution to soft tissue artefact and impact on knee joint kinematics.* Journal of Biomechanics
BOGO (2016) *Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image.* In Computer Vision–ECCV
BOLAÑOS (2021) *A three-dimensional virtual mouse generates synthetic training data for behavioral analysis.* Nature methods
CAO (2019) *OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields.* IEEE transactions on pattern analysis and machine intelligence
DONG (2021) *Fast and robust multi-person 3d pose estimation and tracking from multiple views.* IEEE Transactions on Pattern Analysis and Machine Intelligence
FANG (2017) *RMPE: Regional Multi-person Pose Estimation.* IEEE International Conference on Computer Vision (ICCV)
FERRARI (2010) *A new formulation of the coefficient of multiple correlation to assess the similarity of waveforms measured synchronously by different motion analysis protocols.* Gait & posture
GORTON (2009) *Assessment of the kinematic variability among 12 motion analysis laboratories.* Gait and Posture
HARNETT (2022) *Validating an inertial measurement unit for cricket fast bowling: a first step in assessing the feasibility of diagnosing back injury risk in cricket fast bowlers during a tele-sport-and-exercise medicine consultation.* PeerJ
KARASHCHUK (2021) *Anipose: a toolkit for robust markerless 3D pose estimation.* Cell reports
KANKO (2021) *Concurrent assessment of gait kinematics using marker-based and markerless motion capture.* Journal of Biomechanics
KESSLER (2019) *A Direct Comparison of Biplanar Videoradiography and Optical Motion Capture for Foot and Ankle Kinematics.* Front. Bioeng. Biotechnol
LEBOEUF (2019) *The effect on conventional gait model kinematics and kinetics of hip joint centre equations in adult healthy gait.* Journal of Biomechanics
LOPER (2015) *SMPL: a skinned multi-person linear model.* ACM Transactions on Graphics
LIU (2022) *Auto calibration of multi-camera system for human pose estimation.* IET Computer Vision
MAHMOOD (2019) *AMASS: Archive of Motion Capture as Surface Shapes.* ICCV
MATHIS (2018) *DeepLabCut: markerless pose estimation of user-defined body parts with deep learning.* Nature Neuroscience
MATTHIS (2022) *FreeMoCap: A free, open source markerless motion capture system*, Zenodo
MOESLUND (2001) *Review - A Survey of Computer Vision-Based Human Motion Capture.* Computer Vision and Image Understanding
NEEDHAM (2021) *The accuracy of several pose estimation methods for 3D joint centre localisation.* Scientific reports
NEEDHAM (2022) *The development and evaluation of a fully automated markerless motion capture workflow*, Journal of Biomechanics.
OSMAN (2020) *Star: Sparse trained articulated human body regressor.* ECCV
PAGNON (2021) *Pose2Sim: An End-to-End Workflow for 3D Markerless Sports Kinematics—Part 1: Robustness.* Sensors
PAGNON (2022a) *Pose2Sim: An End-to-End Workflow for 3D Markerless Sports Kinematics—Part 2: Accuracy.* Sensors
PAGNON (2022b) *Pose2Sim: An open-source Python package for multiview markerless kinematics.* Journal of Open Source Software
PAGNON (2022c) *A 3D markerless protocol with action cameras – Key performance indicators in boxing.* ECSS
PEREIRA (2022) *SLEAP: A deep learning system for multi-animal pose tracking.* Nature methods
PETIT (2010) *Multicamera real-time 3d modeling for telepresence and remote collaboration*, International journal of digital multimedia broadcasting
RAJAGOPAL (2015) *Full-Body Musculoskeletal Model for Muscle-Driven Simulation of Human Gait.* IEEE Transactions on Biomedical Engineering
REMPE (2020) *Contact and Human Dynamics from Monocular Video.* ECCV
SENGUPTA (2020) *Synthetic training for accurate 3d human pose and shape estimation in the wild.* arXiv
UHLRICH (2022) *OpenCap: 3D human movement dynamics from smartphone videos.* biorXiv
WANG (2022) *YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors.* arXiv
WOOD (2021) *Fake it till you make it: face analysis in the wild using synthetic data alone.* IEEE/CVF

Smallest worthwhile change

Quantifying accuracy requirements

First, reasonable results as compared to known marker-based errors:

9° errors vs gold standard [Kessler 2019]
10° inter-operator errors [Gorton2009]
3° soft tissue artifacts [Benoit2015]
3° joint position errors [Leboeuf2019]

Is the markerless protocol accurate enough to detect SWC?

$\boxed{\text{SWC} = 0.2 \times \text{SD}}$

Mean inter-protocol differences

marker-based vs. markerless with GoPros

- Knee flexion Mean_{err} = 3,3°
- Pelvis rotation Mean_{err} = 4.9°
- Shoulder Mean_{err} = 1.1°

Between-subjects SWC:

→ More than enough to differentiate athletes

Within-subject SWC:

Very small (experts): Order frame, mm, dm/s, 1°

- Knee flexion SWC = 0.7°
- Pelvis rotation SWC = 0.9°
- Shoulder SWC = 1.1°

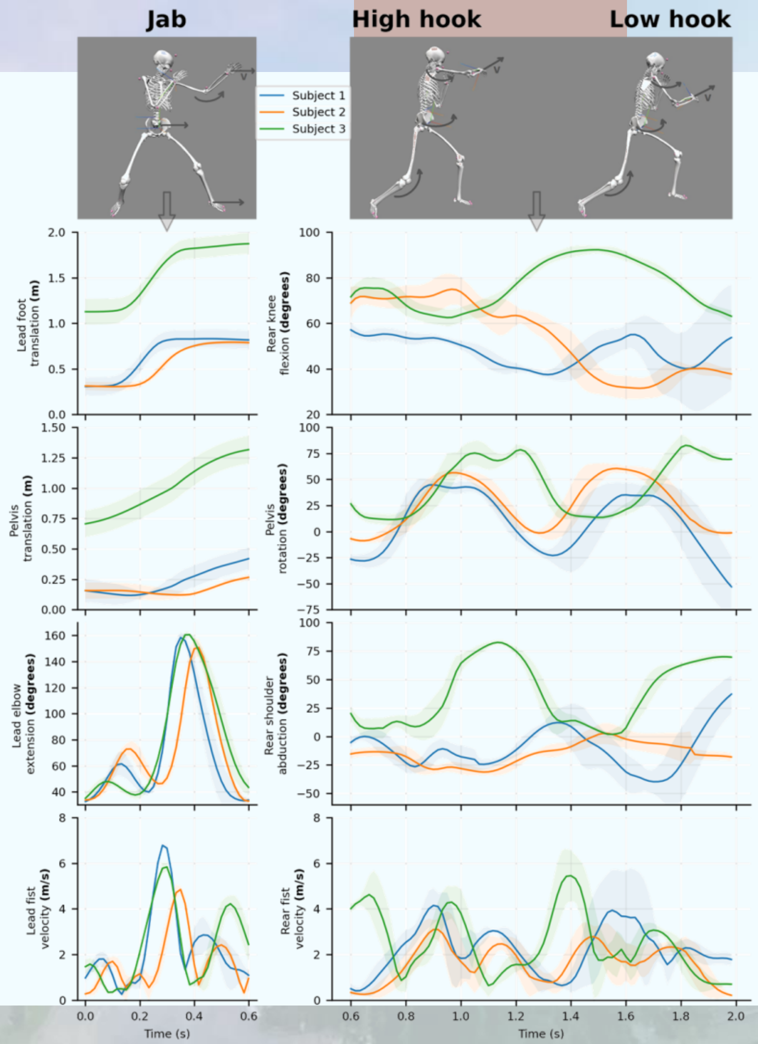
→ Not enough to differentiate trials

Between-subjects SWC

in more normative movements (cricket): [Harnett 2022]

- Knee flexion SWC = 1.9°
- Pelvis rotation SWC = 1.2°
- Shoulder SWC = 3.9°

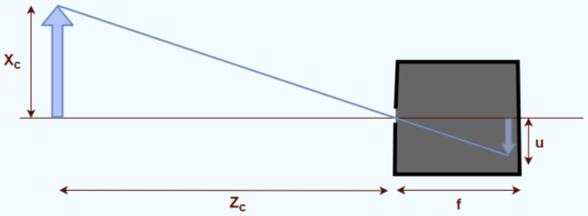
→ Might be enough



Triangulation

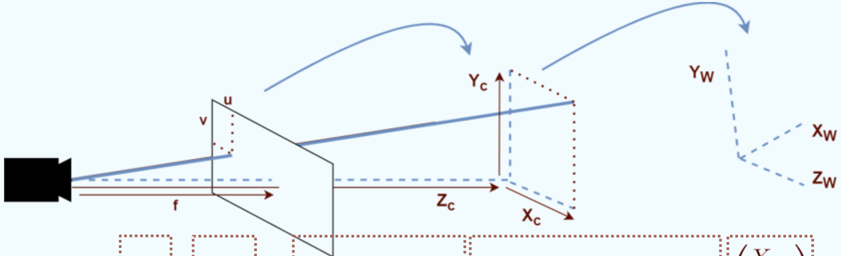
With computer vision

Simplified pinhole model



Thales theorem : $\frac{Z_C}{f} = \frac{X_C}{u} \rightarrow Zc\ u = f\ Xc$

Full 3D pinhole model in world coordinate system:



$$Z_C \times \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_u & 0 & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R}_{3\times3} & \mathbf{T}_{3\times1} \end{pmatrix} \begin{pmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{pmatrix}$$

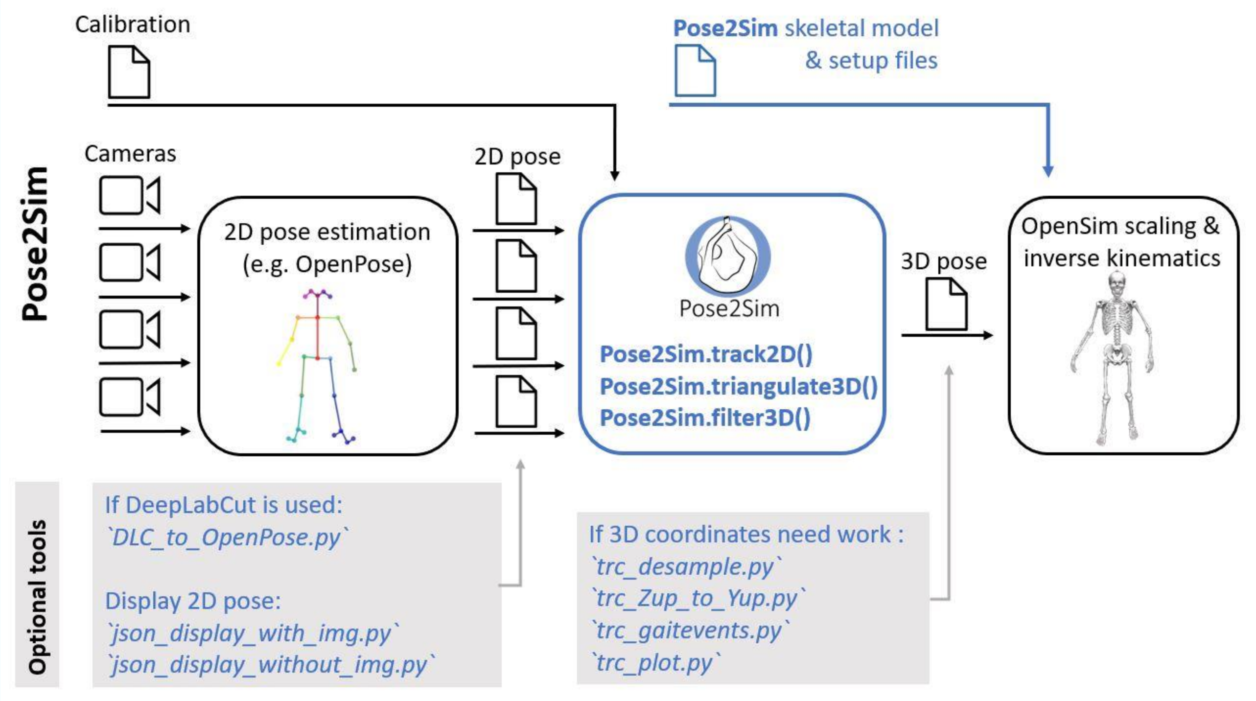
$$Z_C \times \vec{q} = \mathbf{K} \quad \mathbf{H} \quad \overrightarrow{Q_W}$$

$$Z_C \times \vec{q} = \mathbf{P} \quad \overrightarrow{Q_W}$$

Scale factor — Image coordinates — Intrinsic parameters — Extrinsic parameters — World coordinates

Projection matrix

Pose2Sim tools & workflow



Perspectives

Ideal motion capture system? [Atha 1984]

- Accurate → Need for more and better labelled keypoints
- No interference with natural movement → OK
- Simple → Need for user-friendly calibration and synchronization + GUI
- Quick set up → OK and analysis → Bottleneck: 2D pose
- Cheap → 5,000 € vs. 100,000 €

Accuracy: More and better labelled keypoints?

Build a new dataset with minimal labeling effort

1. Gather numerous sports marker files
2. Fit an SMPL shape [Mahmood 2019]
3. Add clothing, scene, cameras. Change shape parameters. Add equipment [Bolanos 2021, Wood 2021]
4. Label keypoints or segment shapes **on one frame** (SMPL: constant topology)

Preprint published yesterday

Better accuracy when using

- MMPose pose detector with Halpe dataset
- Optimization of parameters of implicit function

Simplicity: Use in training, not only in research?

Enhance Pose2Sim

1. Simplify synchronization procedure
2. Simplify calibration (on a person's limb dimensions like with wand) [Liu 2022]
3. Support multi-person analysis [Dong 2019]
4. Build GUI
5. Estimate dynamics

Pose2Sim config & log files

Pose2Sim:

- Triangulates with **weighted DLT**
- Automatically chooses **the right person** to triangulate
- Is robust to **person exiting** and entering camera field
- Offers **flexible settings** for triangulation and filtering
- Gives **precise feedback**: which keypoints, cameras, and frames pose problem?

Config file

```
###############################################################################
## PROJECT PARAMETERS                                                        ##
###############################################################################

# Configure your project parameters here

[project]
project_dir = '' # BETWEEN SINGLE QUOTES! # If empty, project dir is current dir
frame_range = [] #For example [10,300], or [] for all frames
frame_rate = 60 #Hz

rawImg_folder_name = 'raw-2d'
calib_folder_name = 'calib-2d'
pose_folder_name = 'pose-2d'
pose_json_folder_extension = 'json'
pose_img_folder_extension = 'img'
poseTracked_folder_name = 'pose-2d-tracked'
pose3d_folder_name = 'pose-3d'
opensim_folder_name = 'opensim'

[pose-2d]
pose_model = 'BODY_25B' #CUSTOM, BODY_25B, BODY_25, BODY_135, BLAZEPOSE,
# HALPE_26, HALPE_68, HALPE_136, COCO_133, COCO, MPII are available,
# from DeepLabCut, OpenPose, MediaPipe BlazePose, and AlphaPose
# See Pose2Sim\skeleton.py for their skeleton hierarchy

[calibration]
type = 'qca' # 'qca', 'checkerboard', 'arucoboard', or 'charucoboard'
  [calibration.qca]
  binning_factor = 1 # Usually 1

  [calibration.checkerboard]
  corners_nb = [7,12] # [H,W] rather than [w,h]
  square_size = 80 # mm # [h,w] if square is actually a rectangle
  frame_for_origin = -1 # starting from zero. -1 if board is at origin on last frame
  show_corner_detection = false # /!\ Beware that corners must be detected on all frames,
  # or else extrinsic parameters may be wrong. Set show_corner_detection to 1 to verify
  from_vid_or_img = 'img' # 'vid' or 'img'
  vid_snapshot_every_N_frames = 100
  vid_extension = 'mp4'
  img_extension = 'jpg' # 'png', 'jpg', etc
```

```
[2d-tracking]
tracked_keypoint = 'Neck' # If the neck is not detected by the pose_model, check skeleton.py
# and choose a stable point for tracking the person of interest (e.g., 'right_shoulder' with BLAZEPOSE)
error_threshold_tracking = 20 # px

[3d-triangulation]
error_threshold_triangulation = 20 # px
likelihood_threshold = 0.3
min_cameras_for_triangulation = 2
interpolation = 'cubic' #linear, slinear, quadratic, cubic, or none
# 'none' if you don't want to interpolate missing points
show_interp_indices = false # true or false (lowercase). For each keypoint, return the frames that had to be interpolated

[3d-filtering]
type = 'butterworth' # butterworth, butterworth_on_speed, gaussian, LOESS, median
display_figures = 'True' # 'True' or 'False'

  [3d-filtering.butterworth]
  type = 'low'
  order = 4
  cut_off_frequency = 10 # Hz
  [3d-filtering.butterworth_on_speed]
  type = 'low'
  order = 4
  cut_off_frequency = 10 # Hz
  [3d-filtering.gaussian]
  sigma_kernel = 2 #px
  [3d-filtering.LOESS]
  nb_values_used = 30 # = fraction of data used * nb frames
  [3d-filtering.median]
  kernel_size = 9

[opensim]
```

Log file

```
---------------------------------------------------------------------
Tracking of the person of interest for Example, for frames 0 to 15.
---------------------------------------------------------------------

--> Mean reprojection error for RHip on frames 0-15 is 0.8 px, which roughly corresponds to 1.1 mm.
--> In average, 0.4 cameras had to be excluded to reach the demanded 30 px error threshold.
Tracking took 0.05 s.
Tracked json files are stored in ~\pose-2d-tracked.


---------------------------------------------------------------------
Triangulation of 2D points for Example, for frames 0 to 15.
---------------------------------------------------------------------

Mean reprojection error for RHip is 1.4 px (~ 0.002 m), reached with 1.0 excluded cameras.
Frames [2,3,4] had to be interpolated.
Mean reprojection error for LHip is 1.2 px (~ 0.002 m), reached with 2.1 excluded cameras.
Frames [] had to be interpolated.
Mean reprojection error for RKnee is 1.6 px (~ 0.002 m), reached with 1.8 excluded cameras.
Frames [3,4,10,11] had to be interpolated.
...
--> Mean reprojection error for all points on frames 0-15 is 1.4 px, which roughly corresponds to 2.1 mm.
Cameras were excluded if likelihood was below 0.4 and if the reprojection error was above 30 px.
In average, 1.68 cameras had to be excluded to reach these thresholds.
Camera 1 was excluded 17% of the time, Camera 2: 8%, Camera 3: 4%, Camera 4: 2%.
Triangulation took 0.47 s.
3D coordinates are stored at ~\pose-3d\Example_0-15.trc.


---------------------------------------------------------------------
Filtering 3D coordinates for Example, for frames 0 to 15.
---------------------------------------------------------------------

--> Filter type: Butterworth low-pass. Order 4, Cut-off frequency 10 Hz.
Filtered 3D coordinates are stored at ~\pose-3d\Example_0-15_filt.trc.
```

Adapted from:

- Presentation template by Slidesgo
- Icons by Flaticon and David Pagnon
- Infographics by Freepik and David Pagnon
- Images created by Freepik and David Pagnon
- Author introduction slide photo created by Freepik
- Text & Image slide photo created by Freepik.com