The Journal of Open Source Software

Sports2D: Compute 2D joint and segment angles from your smartphone

David Pagnon [1¶]

1 Centre for the Analysis of Motion, Entertainment Research & Applications (CAMERA), University of Bath, Claverton Down, Bath, BA2 7AY, UK ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 17 February 2024
Published: unpublished

Summary

Sports2D provides a user-friendly solution for automatic analysis of human movement from a video. This Python package uses 2D markerless pose estimation to detect joint coordinates from videos, and then computes 2D joint and segment angles. It can be installed either locally or on a free server, which makes it possible to run it directly from a smartphone.

The output incorporates annotated videos and image sequences overlaid with joint locations, joint angles, and segment angles, for each of the detected persons. This information is also stored in .csv files for further analysis, editable on MS Excel® or any other spreadsheet editor.

Sports2D may be useful for clinicians as a decision supports system (CDSS) (Bright et al., 2012), as well as for gait analysis (Whittle, 2014) or ergonomic design (Patrizi et al., 2016). Sports coaches can also use it to quantify key performance indicators (KPIs) (O'Donoghue, 2008; Pagnon, Domalain, Robert, et al., 2022), or to better understand, correct, or compare athletes' movement patterns. Finally, it can be used by researchers as a simple tool for 2D biomechanical analysis on the fly. One of the multiple use cases would be to evaluate ACL injury risks from deceleration drills (Di Paolo et al., 2021).

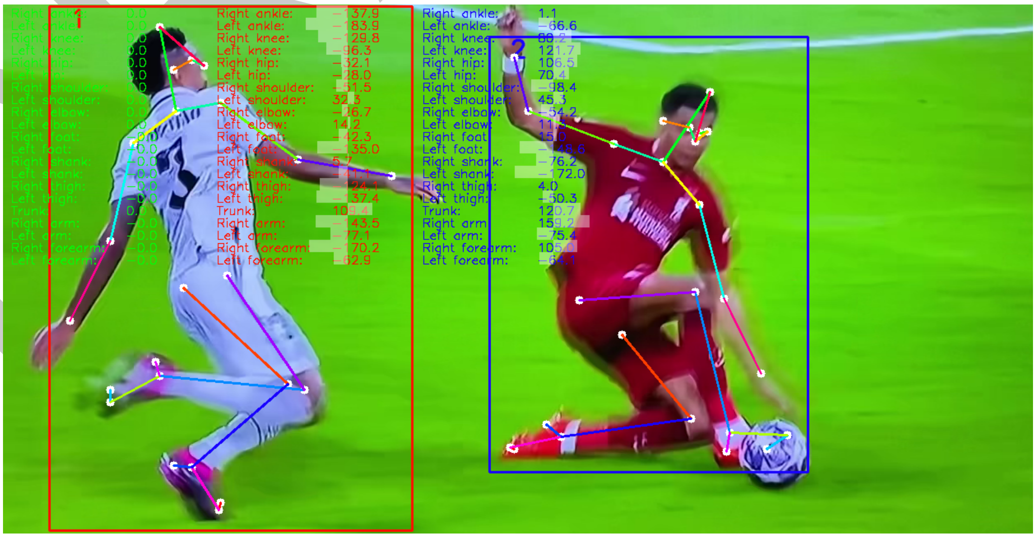

Figure 1: Example results from a demonstration video.

The Journal of Open Source Software

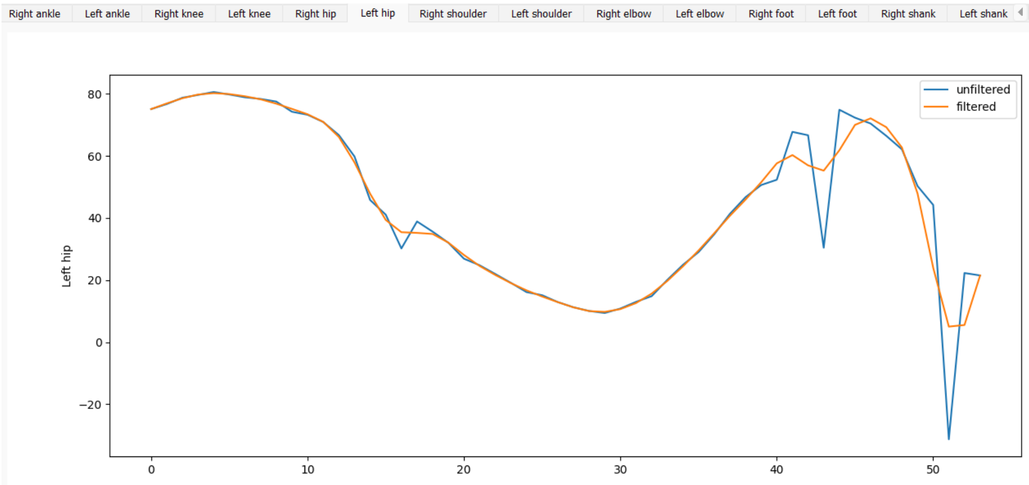

Figure 2: Example plot of joint angle evolution.

Statement of need

Machine learning has recently accelerated the development and availability of markerless kinematics (Colyer et al., 2018; Zheng et al., 2023), which allows for the collection of kinematic data without the use of physical markers or manual annotation.

A large part of these tools focus on 2D analysis, such as `OpenPose` (Cao et al., 2019), `BlazePose` (Bazarevsky et al., 2020), or `DeepLabCut` (Mathis et al., 2018). Although they bear the advantage of being open-source, they are not easily accessible to people who do not have a programming background, and the output is not directly usable for further kinematic investigation. Yet, clinical acceptance of new technologies is known to be influenced not only by their price value and their performance, but also by their perceived ease-of-use, the social influence around the customer, and other parameters described by the Unified Theory of Acceptance and Use of Technology (UTAUT2) (Venkatesh et al., 2012).

In fact, there is a clear trade-off between accuracy and ease-of-use. Some open-source tools focus on the accuracy of a 3D analysis by using multiple cameras, such as `Pose2Sim` (Pagnon, Domalain, & Reveret, 2022) or `OpenCap` (Uhlrich et al., 2022). These, however, require either a certain level of programming skills, or a particular hardware setup. Some other tools choose to put more emphasis on user-friendliness, and point out that 2D analysis is often sufficient when the analyzed motion mostly lies in the sagittal or frontal plane. `Sit2Stand` (Boswell et al., 2023) and `CP GaitLab` (Kidziński et al., 2020) provide such tools, focused however on very specific tasks. `Kinovea` (Kinovea), on the other hand, is a widely used software for sports performance analysis, which provides multiple additional features. However, it relies on tracking manual labels. This can be time-consuming when analyzing numerous videos, and it may also be lacking robustness when the tracked points are lost. It is also only available on Windows, and requires the user to transfer files prior to analysis.

`Sports2D` is an alternative solution that aims at filling this gap: it is free and open-source, easy to install, can be run from any smartphone or computer, and automatically provides 2D joint and segment angles without the need for manual annotation. It is also robust, and can be used to analyze numerous videos at once. The motion of multiple people can be analyzed in the same video, and the output is directly usable for further statistical analysis.

The Journal of Open Source Software

Workflow

`Sports2D` can be installed and run two different ways: locally, or on a Google Colab® free server (Bisong & Bisong, 2019).

- *If run locally*, it is installed under Python via `pip install sports2d`. Two options are then offered: either run it with BlazePose (Bazarevsky et al., 2020) as a pose estimation model, or with OpenPose (Cao et al., 2019). BlazePose comes preinstalled and is very fast to run, however it is less accurate and only detects one person per video. OpenPose is more accurate, allows for the detection of multiple people, and comes with more fine-tuning, but it is slower and requires the user to install it themselves.
- *If run on Colab*, it can be installed in one click from any computer or smartphone device, either every time the user needs it, or once for all on Google Drive®. In both cases, OpenPose is automatically installed and runs by default, and video and table results are automatically saved on Google Drive®. A video tutorial can be found at this address: https://www.youtube.com/watch?v=Er5RpcJ8o1Y.

After installation, the user can choose one or several videos to analyze.`Sports2D` then goes through two stages:

- **Pose detection:** Joint centers are detected for each video frame. If OpenPose is used, multiple persons can be detected with consistent IDs across frames. A person is associated to another in the next frame when they are at a small Euclidian distance. Sequences of missing data are interpolated if they are less than N frames long, N being a threshold defined by the user. Resulting coordinates can be filtered with a Butterworth (Butterworth, 1930), Gaussian, Median, or LOESS (Cleveland, 1981) filter. They can also be plotted. Note that locations are in pixels, but can be converted to meters if the user provides the distance between two points in the video.
- **Joint and segment angle estimation:** Specific joint and segment angles can be chosen, and are computed from the previously calculated positions. Angles are consistent regardless of the direction the participant is facing: the participant is considered to go to the left when their toes are to the left of their heels, and to the right otherwise. Resulting angles can be filtered in the same way as point coordinates, and they can also be plotted.

Joint angle conventions are as follows (Figure 3):

- Ankle dorsiflexion: Between heel and big toe, and ankle and knee;
- Knee flexion: Between hip, knee, and ankle;
- Hip flexion: Between knee, hip, and shoulder;
- Shoulder flexion: Between hip, shoulder, and elbow;
- Elbow flexion: Between wrist, elbow, and shoulder.

Segment angles are measured anticlockwise between the horizontal and the segment lines:

- Foot: Between heel and big toe;
- Shank: Between knee and ankle;
- Thigh: Between hip and knee;
- Arm: Between shoulder and elbow;
- Forearm: Between elbow and wrist;
- Trunk: Between shoulder midpoint and hip midpoint.

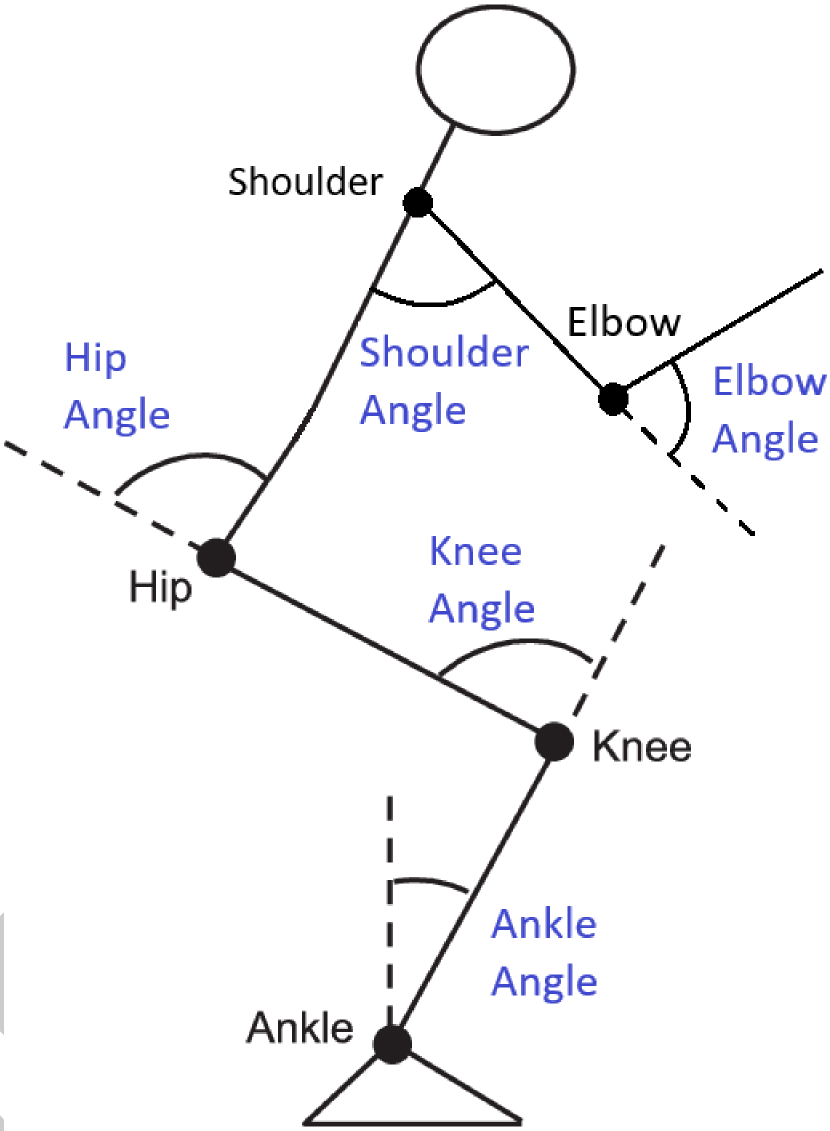

Figure 3: Joint angle conventions. Adapted from (Yang et al., 2007).

Limitations

The user of `Sports2D` should be aware of the following limitations:

- Results are acceptable only if the participants move in the 2D plane, from right to left or from left to right. If you need research-grade markerless joint kinematics, consider using several cameras, and constraining angles to a biomechanically accurate model. See `Pose2Sim` (Pagnon, Domalain, & Reveret, 2022) for example.
- Angle estimation is only as good as the pose estimation algorithm, i.e., it is not perfect (Wade et al., 2022), especially if motion blur is significant such as on some broadcast videos.
- Google Colab does not follow the European GDPR requirements regarding data privacy (Minssen et al., 2020). Install locally if this matters.

Acknowledgements

I would like to acknowledge Rob Olivar, a sports coach who enlightened me about the need for such a tool.
I also acknowledge the work of the dedicated people involved in the many major open-source software programs and packages used by `Sports2D`, such as `Python`, `OpenPose`, `BlazePose`, `OpenCV` (Bradski, 2000), among others.

References

Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., & Grundmann, M. (2020). Blazepose: On-device real-time body pose tracking. *arXiv Preprint arXiv:2006.10204*. https://doi.org/10.48550/arXiv.2006.10204

Bisong, E., & Bisong, E. (2019). Google colaboratory. *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*, 59–64. https://doi.org/10.1007/978-1-4842-4470-8

Boswell, M. A., Kidziński, Ł., Hicks, J. L., Uhlrich, S. D., Falisse, A., & Delp, S. L. (2023). Smartphone videos of the sit-to-stand test predict osteoarthritis and health outcomes in a nationwide study. *Npj Digital Medicine*, *6*(1), 32. https://doi.org/10.1038/s41746-023-00775-1

Bradski, G. (2000). The OpenCV library. *Dr. Dobb's Journal of Software Tools*.

Bright, T. J., Wong, A., Dhurjati, R., Bristow, E., Bastian, L., Coeytaux, R. R., Samsa, G., Hasselblad, V., Williams, J. W., Musty, M. D., & others. (2012). Effect of clinical decision-support systems: A systematic review. *Annals of Internal Medicine*, *157*(1), 29–43. https://doi.org/10.7326/0003-4819-157-1-201207030-00450

Butterworth, S. (1930). On the theory of filter amplifiers. *Wireless Engineer*, *7*(6), 536–541.

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2019). OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(1), 172–186. https://doi.org/10.1109/TPAMI.2019.2929257

Cleveland, W. S. (1981). LOWESS: A program for smoothing scatterplots by robust locally weighted regression. *American Statistician*, *35*(1), 54. https://doi.org/10.2307/2683591

Colyer, S. L., Evans, M., Cosker, D. P., & Salo, A. I. (2018). A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system. *Sports Medicine-Open*, *4*(1), 1–15. https://doi.org/10.1186/s40798-018-0139-y

Di Paolo, S., Zaffagnini, S., Tosarelli, F., Aggio, F., Bragonzoni, L., Grassi, A., & Della Villa, F. (2021). A 2D qualitative movement assessment of a deceleration task detects football players with high knee joint loading. *Knee Surgery, Sports Traumatology, Arthroscopy*, *29*, 4032–4040. https://doi.org/10.1007/s00167-021-06709-2

Kidziński, Ł., Yang, B., Hicks, J. L., Rajagopal, A., Delp, S. L., & Schwartz, M. H. (2020). Deep neural networks enable quantitative movement analysis using single-camera videos. *Nature Communications*, *11*(1), 4054. https://doi.org/10.1038/s41467-020-17807-z

Kinovea. *Kinovea - a microscope for your videos*. https://www.kinovea.org/features.html; GitHub. https://www.kinovea.org/features.html

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., & Bethge, M. (2018). DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, *21*(9), 1281–1289. https://doi.org/10.1038/s41593-018-0209-y

Minssen, T., Rajam, N., & Bogers, M. (2020). Clinical trial data transparency and GDPR compliance: Implications for data sharing and open innovation. *Science and Public Policy*, *47*(5), 616–626. https://doi.org/10.2139/ssrn.3413035

O'Donoghue, P. (2008). Principal components analysis in the selection of key performance indicators in sport. *International Journal of Performance Analysis in Sport*, *8*(3), 145–155. https://doi.org/10.1080/24748668.2008.11868456

Pagnon, D., Domalain, M., & Reveret, L. (2022). Pose2Sim: An open-source python package for multiview markerless kinematics. *Journal of Open Source Software*, *7*(77), 4362. https://doi.org/10.21105/joss.04362

Pagnon, D., Domalain, M., Robert, T., Lahkar, B.-K., Moussa, I., Saulière, G., Goyallon, T., & Reveret, L. (2022). A 3D markerless protocol with action cameras – Key performance indicators in boxing. *2022 Congress of the European College of Sport Science (ECSS)*. https://hal.archives-ouvertes.fr/hal-03790926

Patrizi, A., Pennestrì, E., & Valentini, P. P. (2016). Comparison between low-cost marker-less and high-end marker-based motion capture systems for the computer-aided assessment of working ergonomics. *Ergonomics*, *59*(1), 155–162. https://doi.org/10.1080/00140139.2015.1057238

Uhlrich, S. D., Falisse, A., Kidziński, Ł., Muccini, J., Ko, M., Chaudhari, A. S., Hicks, J. L., & Delp, S. L. (2022). *OpenCap: 3D human movement dynamics from smartphone videos*. 2022.07.07.499061. https://doi.org/10.1101/2022.07.07.499061

Venkatesh, V., Thong, J. Y., & Xu, X. (2012). Consumer acceptance and use of information technology: Extending the unified theory of acceptance and use of technology. *MIS Quarterly*, 157–178. https://doi.org/10.2307/41410412

Wade, L., Needham, L., McGuigan, P., & Bilzon, J. (2022). Applications and limitations of current markerless motion capture methods for clinical gait biomechanics. *PeerJ*, *10*, e12995. https://doi.org/10.7717/peerj.12995

Whittle, M. W. (2014). *Gait analysis: An introduction*. Butterworth-Heinemann.

Yang, Y., Baker, M., Graf, S., Larson, J., & Caiozzo, V. J. (2007). Hypergravity resistance exercise: The use of artificial gravity as potential countermeasure to microgravity. *Journal of Applied Physiology*, *103*(5), 1879–1887. https://doi.org/10.1152/japplphysiol.00772.2007

Zheng, C., Wu, W., Chen, C., Yang, T., Zhu, S., Shen, J., Kehtarnavaz, N., & Shah, M. (2023). Deep learning-based human pose estimation: A survey. *ACM Computing Surveys*, *56*(1), 1–37. https://doi.org/10.1145/3603618