Fritz Webering*, Leili Emamzadehei, and Holger Blume

Modeling and Error Analysis in Camera-Based Jump Height Measurement

https://doi.org/10.1515/cdbme-2022-1159

Abstract: ***Introduction:*** In this work, we use simulated data to quantify the different failure mechanisms of a previously presented low-cost jump height measurement system, based on widely available consumer smartphone technology. ***Methods:*** In order to assess the importance of the different preconditions of the jump height measurement algorithm, we generate a synthetic dataset of 2000 random jump parabolas for 2000 randomly generated persons without real-world artifacts. We then selectively add different perturbations to the parabolas and reconstruct the jump height using the evaluated algorithm. The degree to which the manipulations influence the reconstructed jump height gives us insights into how critical each precondition is for the method's accuracy. ***Results:*** For a subject-to-camera distance of 2.5 meters, we found the most important influences to be tracking inaccuracies and distance changes (non-vertical jumps). These are also the most difficult factors to control. Camera angle and lens distortion are easier to handle in practice and have a very low impact on the reconstructed jump height. The intraclass correlation value ICC(3,1) between true jump height and the reconstruction from distorted data ranges between 0.999 for mild and 0.988 for more severe distortions. ***Conclusion:*** Our results support the design of future studies and tools for accurate and affordable jump height measurement, which can be used in individual fitness, sports medicine, and rehabilitation applications.

Keywords: vertical jump height, sports, camera calibration, gravity, parabola, simulation

1 Introduction and Related Work

The assessment of vertical jump height is an important tool to gauge ballistic lower body strength in sports sciences and sports medicine. Vertical jump height, as defined by Bobbert and van Ingen [1], is the maximum vertical movement of the body's center of mass. However, determining the exact center of mass is not straightforward because it depends on the movement of all body parts, including flexible tissue [2].

***Corresponding author: Fritz Webering, Holger Blume,** Institute of Microelectronic Systems, Leibniz University Hannover, Appelstr. 4, 30167 Hannover, Germany, e-mail: {webering,blume}@ims.uni-hannover.de
Leili Emamzadehei, Leibniz University Hannover, Germany

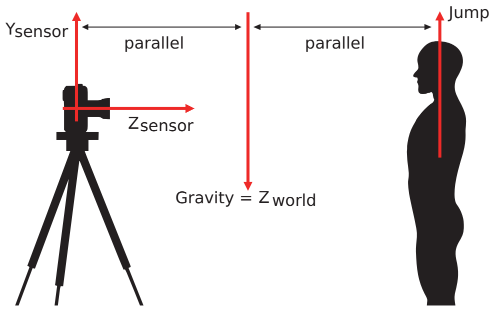

Fig. 1: Illustration of the jump height measurement system and preconditions: The jump must be straight up. The camera must be still (e.g. on a tripod) and aligned vertically. A typical modern smartphone camera meets the frame rate and resolution requirements and contains an accelerometer which helps with alignment.

Because of this difficulty, most practical jump height measurement methods determine the height indirectly from a different measured quantity, such as: Motion capturing, force plates, floor contact detectors, frame counting in high-speed videos, or trajectory analysis in high-speed video.

In this work, we take a closer look at the jump height measurement method proposed by Webering et al. [3], which analyzes the vertical trajectory of the subject in a slow-motion video taken, for example, with a modern smartphone. The algorithm uses the curvature of the free-fall parabola to perform a partial camera calibration, calculating only the pixel-per-meter ratio at the specific object-space plane where the subject is jumping. This work aims to examine the different failure mechanisms of the algorithm using synthetic data in order to quantify the limits of its accuracy under real-world conditions.

All results in this paper and code used to generate those results are available under DOI:10.25835/s067v1ho.

2 Problem Statement

The algorithm proposed by Webering et al. [3] is based on the assumption that the mapping from the Z_{world} axis—and thus the motion of the person jumping in the real world—to the Y_{sensor} axis—meaning, the motion of the person jumping in the video—is linear. The existence of this linear mapping is tied to four main assumptions, as illustrated in Fig. 1:

1. That the camera is aligned upright, parallel to gravity,
2. that the person jumps straight up,

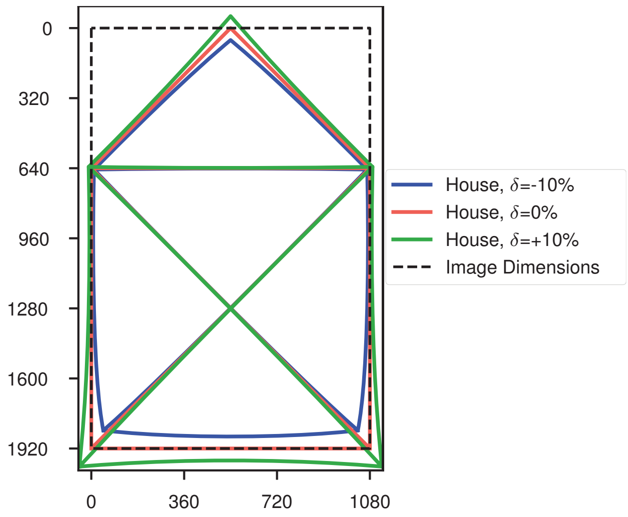

Fig. 2: Visualization how the minimum and maximum lens distortion values used in this study affect a house shape in a 1920x1080 px image. All axes are in pixel coordinates. The maximum deflection in the corners is 94.7 px.

3. that there is no lens distortion,
4. and that the center of mass can be tracked accurately.

Assumption 3 can be easily solved with a checkerboard calibration for the camera in question. Additionally, Webering et al. [3] mention that smartphones apparently ship with lens correction built-in, at least the models used in the study. However, the other assumptions are less clear:

Assumption 1 is addressed in [3] by manually positioning the smartphone so that one axis of the built-in accelerometer points directly in the direction of gravity. This approach only works if the axes of the built-in accelerometer are parallel to image sensor plane and if the operator can align the smartphone accurately on a suitably stable mount.

Assumption 2 requires that the subject is skilled enough to jump perfectly straight up, which may not always be true, especially if they focus on jumping as high as possible, as per test instructions.

Finally, assumption 4 is probably the most difficult one to satisfy because the center of mass (CoM) is just a mathematical construct that depends on the mass distribution in the subject's body. In [3], a fiducial marker is applied to the subject's back near the CoM, and in [4], the CoM location is estimated using OpenPose [5].

Violating any of these preconditions will certainly result in inaccurate jump height results. However, the isolated impact of every individual assumption is unknown. Thus, the aim of this work is to analyze the failure modes of the assumptions 1 through 4 numerically, using synthetic data, in order to judge the accuracy requirements for equipment, operator, and subject, as well as the effect of tracking inaccuracies.

3 Modeling and Evaluation

In order to model each assumption from section 2 individually, we generated a synthetic dataset of N perfect free-fall trajectories without any real-world artifacts. $N = 2000$ yielded acceptably low result variation with a simulation time below 1 hour. The random synthetic jump heights are drawn from the squat jump height distributions in [6], and the standing height of the CoM is based on the body height distribution taken from [7]. These perfect parabolas are projected into a 2-dimensional image coordinate system using the python package CameraTransform [8] without rounding to integer pixels.

The camera model was constructed to imitate the smartphone camera in front of the subject used in [3], with a resolution of 1920x1080 px at 240 FPS and a vertical field of view of 63° in portrait orientation. The default distance between camera and subject was 2.5 m with a camera height of 1 m, because this fits the whole subject into the image during the jump, and because [3] used a similar setup. After this step, the jump height can be reconstructed using the algorithm by Webering et al. [3] with perfect accuracy.

We can then selectively add perturbations to the data, intentionally violating the assumptions mentioned above, and measure the impact on the reconstructed jump heights. The perturbations parameters are:

- **Noise amplitude** A. In some tests (see Section 4), noise was added to the pixel coordinates after camera mapping. Three different types of noise were evaluated: White uniform noise, white Gaussian noise with $A = 1.96\sigma$ (such that 95 % of samples lie within A), and fractal noise where the amplitude of the lowest octave is equal to A.
  Since the impact of the fractal noise on the accuracy was greater than the other two types of noise, it was also used in the distance evaluation.
- **Initial camera-to-subject distance** d. This determines the scale of the parabola in the image. Since scale alone causes no degradation in the jump height estimation, we added fractal noise to simulate tracking errors. As an additional case, the tracked points were rounded to whole pixels, which is equivalent to a shaped noise with $A = 0.5$ px.
- **Camera pitch angle** α. A negative angle means that the camera is looking down; a positive angle represents looking up. The pitch angle adds perspective distortion to the shape of the parabola in the image coordinate system.
- **Relative subject movement** Δd. The change in distance d during the jump. A negative value means jumping to-

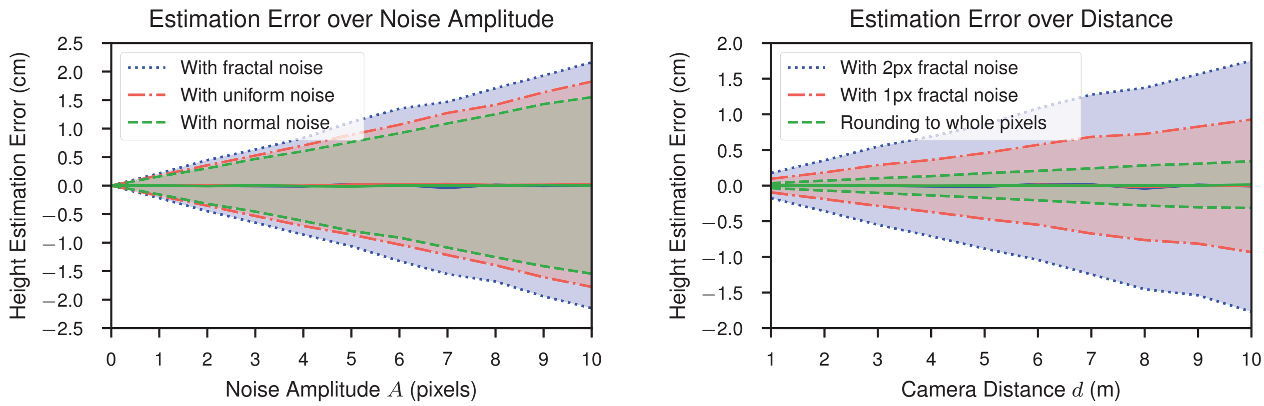

Fig. 3: Results for jump height estimation for the parameters A and d (see Section 3). Solid lines are the result values. Shaded areas between the dashed lines are the 95 % limits of agreement (LOA), which contain 95 % of the data points. This is equal to 1.96 times the standard deviation of the height estimation errors for the specific parameter value on the x-axis.

ward the camera, and a positive value means jumping away. Both change the perspective distortion over time.

– **Brown–Conrady radial lens distortion** δ. The three parameters of the radial distortion model are set to $K_1 = K_2 = K_3 = \delta$. In this work, δ is given in percent, so a value of $\delta = 10\,\%$ means 0.1. Fig. 2 visualizes the highest and lowest values for distortion used. The maximum displacement is 94.7 px in the image corners at $\delta = \pm 10\,\%$.

4 Results and Discussion

Our evaluation shows that the jump height estimation error is linear with d if noise is present. This increase with d is expected because the noise amplitude becomes larger relative to the parabola amplitude in the image as the subject recedes from the camera. At the chosen default camera distance of 2.5 m, with a fractal noise amplitude of 2 px, the 95 % LOA of the height estimation error is ±0.43 cm, as seen in Fig. 3.

A similar conclusion can be drawn for the accuracy of the tracking algorithm—modeled as noise amplitude A—which also shows a linear relationship to the estimation error. For the chosen default camera distance of 2.5 m, the 95 % LOA of the jump height error increases by ±0.22 cm for every 1 px increase in A for fractal noise.

The results for $\Delta d, \alpha$, and δ in Fig. 4 include no added noise to show just the effect of the respective parameter. We observe a systematic error in jump height estimation for these parameters, in contrast to the noise analysis in Fig. 3 where the mean error was always 0.

In the case of the non-vertical jump with $\Delta d \neq 0$, we see an increase in the estimated jump height when the subject jumps away from the camera and a decrease in the opposite direction. This effect of Δd is most likely responsible for the within-subject clustering observed in [3]. The random variation, as represented by the 95 % LOA, increases together with the systematic error. For the default camera distance of 2.5 m, the error is $0.51\,\mathrm{cm}$ for every $10\,\mathrm{cm}$ increase in Δd, and the 95 % LOA increases by 0.25 cm. Placing the camera closer to the subject at $d = 1\,\mathrm{m}$ increases these values significantly to $1.26 \pm 0.61\,\mathrm{cm}$ per $10\,\mathrm{cm}$ of Δd.

For the camera pitch angle α, we observe a similar relationship: Increasing α decreases the systematic jump height estimation error. For the default camera distance of 2.5 m, the systematic error with 95 % LOA is $-0.31 \pm 0.20\,\mathrm{cm}$ for every $10°$ increase in α. With a closer camera at $d = 1\,\mathrm{m}$, the perspective distortion leads to an increased error of these values significantly to $-0.76 \pm 0.49\,\mathrm{cm}$ per $10°$ increase in α in this approximately linear region. Since it is easily possible to achieve alignment accuracy better than 10° with the naked eye, we stipulate that the influence of α can be neglected when an accelerometer or similar tool is used for alignment.

Lens distortion has only a minimal impact at $d = 2.5\,\mathrm{m}$ with an error of $0.01 \pm 0.03\,\mathrm{cm}$ per $10\,\%$ increase of δ in the examined value range and a slightly more significant error of $0.07 \pm 0.23\,\mathrm{cm}$ per 10 percent of δ at $d = 1\,\mathrm{m}$.

When combining all distortions, the ICC(3,1) value for true jump height vs. the height reconstructed from distorted data ranges between 0.999 for the optimistic case $A = \Delta d = \alpha = \delta = 3$ (units: px/cm/degrees/%) and 0.988 for pessimistic distortions $A = \Delta d = \alpha = \delta = 10$.

4.1 Conclusion

From the presented results, we can conclude that not all preconditions of the algorithm are equally significant. Camera

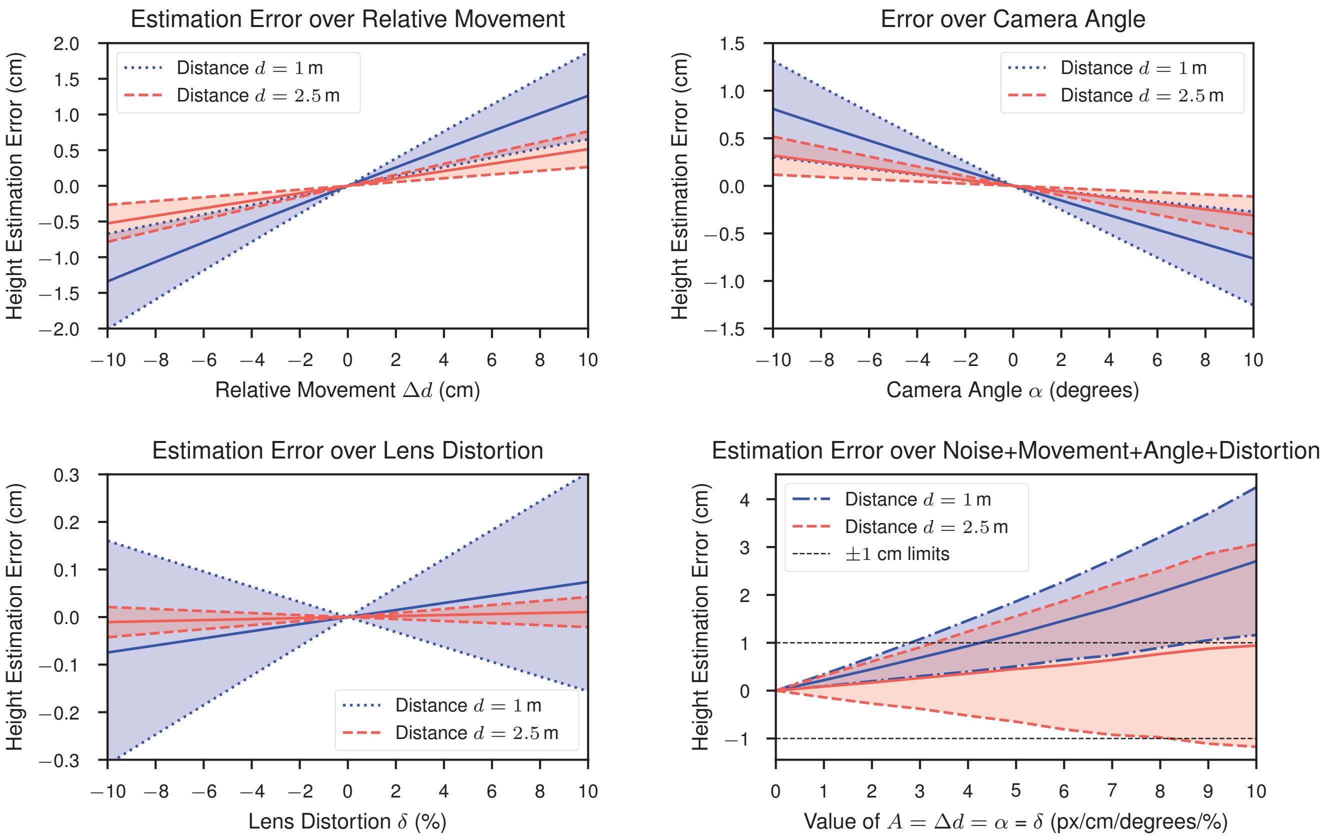

Fig. 4: Results for jump height estimation for the parameters Δd, α, δ and a worst-case combination where all parameters were set to the same numeric value. Solid lines are the result values. Dashed lines and shaded areas are the 95 % LOA (see explanation in Fig. 3).

pitch has a minimal impact for small angles, so sub-degree accuracy is not necessary for accurate jump height results. Lens distortion can be similarly disregarded for devices that provide images with low lens distortion like smartphones.

The most important factors when measuring jump heights using the method from [3] are the verticality of the jump resulting in a low Δd, as well as an accurate method for tracking the CoM in the video.

The findings described in this work can help in developing accurate and low-cost jump height measurement tools for fitness and rehabilitation purposes, using readily available smartphone technology.

Author Statement

The authors state no funding and no conflict of interest.

References

[1] M. F. Bobbert and G. J. van Ingen Schenau, "Coordination in vertical jumping," *Journal of biomechanics*, 1988.

[2] L. F. Aragón, "Evaluation of four vertical jump tests: Methodology, reliability, validity, and accuracy," *Measurement in physical education and exercise science*, vol. 4, no. 4, 2000.

[3] F. Webering, L. Seeger, N. Rother, and H. Blume, "Measuring vertical jump height using a smartphone camera with simultaneous gravity-based calibration," in *IEEE International Conference on Communications and Electronics (ICCE)*, 2021.

[4] F. Webering, H. Blume, and I. Allaham, "Markerless camera-based vertical jump height measurement using openpose," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 3863–3869.

[5] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: realtime multi-person 2d pose estimation using part affinity fields," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, pp. 172–186, 2019.

[6] R. Centeno-Prada, C. López, and J. Naranjo Orellana, "Jump percentile: A proposal for evaluation of high level sportsmen," *The Journal of sports medicine and physical fitness*, vol. 55.

[7] A. Jelenkovic, Y.-M. Hur, R. Sund, Y. Yokoyama et al., "Genetic and environmental influences on adult human height across birth cohorts from 1886 to 1994," *eLife*, vol. 5, Dec. 2016. [Online]. Available: https://doi.org/10.7554/elife.20320

[8] R. C. Gerum, S. Richter, A. Winterl et al., "Cameratransform: A python package for perspective corrections and image mapping," *SoftwareX*, vol. 10, p. 100333, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352711019302018