Review of "Estimating 3D Motion and Forces of Person–Object Interactions from Monocular Video"

Nawel TORCHIO Leos COUTROT

December 8, 2025

Abstract

Reconstructing human–object interactions from visual data is a challenging problem. The paper "Estimating 3D Motion and Forces of Person–Object Interactions from Monocular Video" proposes a method to recover, from a single RGB video, 3D human motion, object motion, contact states, joint torques, and external forces. The approach combines a multi-stage vision pipeline with a physics-based trajectory optimization enforcing rigid-body dynamics, pose priors, and contact constraints.

The method is evaluated on two datasets: a controlled Parkour dataset equipped with motion-capture and force-plate measurements, and a Handtool dataset composed of instructional videos. The authors show that their approach produces accurate 3D poses compared to existing monocular baselines and yields coherent estimates of forces and torques. Overall, the paper demonstrates that physically consistent human–object interactions can be inferred from monocular RGB sequences alone.

1 Introduction and Context

Human–object interactions are central to many everyday tasks and play an important role in robotics, biomechanics, and computer vision. Humans can effortlessly learn actions by observing others, a process often referred to as *behavioral cloning*. Reproducing this ability computationally, however, remains difficult, particularly when only monocular videos are available. These videos lack depth, and object interactions often involve occlusions and ambiguous viewpoints.

Traditional approaches rely on motion capture and force plates, which yield accurate measurements of joints, contacts, and forces, but require controlled laboratory setups and do not scale to the large number of instructional videos available online. Recovering comparable information from standard RGB videos therefore represents a significant challenge.

The paper "Estimating 3D Motion and Forces of Person–Object Interactions from Monocular Video" [Li et al., 2019] addresses this problem by combining image-based detection with a physics-driven optimization framework. The method produces 3D trajectories of both the human and the manipulated object, estimates contact events, and reconstructs external forces and joint torques from monocular video alone.

1.1 Motivation

Monocular video is abundant and easy to capture, but inferring 3D information from a single viewpoint is inherently ambiguous. Objects manipulated by a person may be textureless or thin, and are frequently occluded by the body. Contacts and forces, although essential to understanding manipulation, are not directly visible in RGB images. Motion capture systems provide this information but are expensive, intrusive, and limited to laboratory conditions. The authors aim to extract physically meaningful motion using only monocular inputs, without requiring specialized hardware.

1.2 Problem Formulation

Given an RGB video, the objective is to estimate:

- the 3D human pose over time,
- the 3D pose of the manipulated object,
- the timing and spatial location of contacts,

- the external forces applied at these contacts,
- and the joint torques generating the human motion.

The proposed solution proceeds in two stages. First, a recognition pipeline extracts 2D image observations, such as human joints, object endpoints, and contact states. These visual cues are then used in a trajectory optimization framework that enforces rigid-body dynamics and contact constraints to recover consistent 3D motion.

1.3 Positioning Within Prior Work

Previous monocular human pose estimation methods primarily focus on recovering 3D joint trajectories, without modelling contact forces, torques, or detailed interactions with objects. Other work relies on motion capture or external sensors, reducing applicability outside controlled environments. The contribution of the paper lies in coupling computer vision with physical reasoning: the system reconstructs the underlying dynamics of human–object interactions using only RGB video, through a combination of image-based losses and physics-based constraints.

2 Methodology of the Original Paper

The method consists of two stages: a recognition stage that extracts 2D observations, and a physical estimation stage that reconstructs 3D trajectories consistent with physical constraints and the 2D measurements.

2.1 Recognition Stage: Extracting 2D Observations

The recognition stage gathers all visual information needed by the trajectory optimizer. Human joints are detected in each frame using OpenPose, producing 2D keypoints. Contact states for hands, feet, and knees are inferred by cropping image regions around these joints and classifying them with a ResNet-based binary classifier. The manipulated object is segmented using Mask R-CNN, which is trained *separately for each object category*. From the segmentation masks, the 2D locations of the object endpoints are extracted. Finally, the authors use HMR (Human Mesh Recovery) to initialize a plausible 3D human pose that serves as the starting point for the optimization.

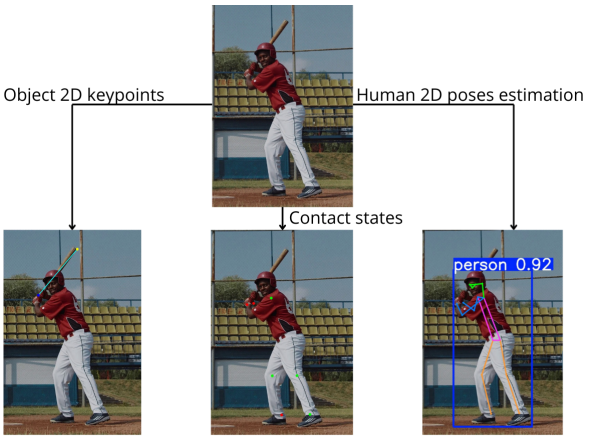

Figure 1: Overview of the Recognition stage

2.2 Physical Estimation Stage: Trajectory Optimization

The second stage estimates 3D human and object motion by solving a constrained trajectory optimization problem over the full sequence. The goal is to find states, contact variables, and forces that satisfy the 2D measurements and the physical laws governing motion.

2.2.1 State and Control Variables

The state vector is

$$x = (q^h, q^o, \dot{q}^h, \dot{q}^o),$$

where q^h and q^o are the generalized coordinates of the human and the object, and $\dot{q}^h$, $\dot{q}^o$ their velocities.

The control vector is

$$u = (\tau_m^h, f_k),$$

with τ_m^h denoting the internal muscle torques (distinct from the total torque which includes contact effects) and f_k the forces applied at contact points.

2.2.2 Optimization Objective

The reconstruction problem is expressed as

$$\min_{x,u,c} \int_0^T l^h(x,u,c) + l^o(x,u,c)\, dt,$$

where l^h and l^o are the human and object loss terms. These terms encourage consistency with the 2D observations, plausible poses, smooth motion, and physically reasonable torques and forces.

2.2.3 Physical and Contact Constraints

Contact Motion Constraint. For detected contacts, the relative position between the human joint and the corresponding object point must be consistent. This is formulated as:

$$\kappa(x,c) = 0 \implies \|p_j^h - p_k^c\| = 0,$$

which implies that the distance between the contact points is zero during the contact phase.

Dynamic Constraint. The motion must satisfy the rigid-body dynamics equation:

$$M(q)\ddot{q} + b(q,\dot{q}) = g(q) + \tau,$$

where $M(q)$ is the mass matrix, $b(q,\dot{q})$ contains Coriolis and centrifugal terms, and $g(q)$ denotes gravitational forces. Note that $\ddot{q}$ represents the joint accelerations.

The function $f(x,c,u)$ appearing in the state space model $\dot{x} = f(x,c,u)$ encodes these dynamics together with contact contributions.

Force Feasibility Constraint. External forces must lie in a feasible set:

$$u \in \mathcal{U}.$$

Object contacts use full 6D wrenches, while ground reaction forces must belong to a friction cone approximated by a polyhedral set.

2.2.4 Cost Terms

The loss terms include components directly defined in the paper:

- **2D reprojection loss:**

$$l_{2D}^h = \sum_j \rho\left(p_j^{2D} - P_{\text{cam}}(p_j(q))\right),$$

  where ρ is a Huber loss and P_{cam} the projection matrix.

- **Pose prior:**

$$l_{\text{pose}}^h = -\log p(q; \text{GMM}),$$

  enforcing anatomically plausible human poses via a Gaussian Mixture Model.

- **Torque regularization:**

$$l_\tau^h = \|\tau_m^h\|^2,$$

  minimizing the energy consumption of the human body.

- **Smoothness term (Motion):**

$$l_{\text{smooth}}^h = \sum_j \left(\|\nu_j(q,\dot{q})\|^2 + \|\alpha_j(q,\dot{q},\ddot{q})\|^2\right),$$

  minimizing spatial velocities ν and accelerations α.

- **Smoothness term (Contact):**

$$l_{\text{smooth}}^c = \omega_k\|\dot{c}_k\|^2 + \gamma_k\|\dot{f}_k\|^2,$$

  regularizing the temporal variation of contact positions and forces.

A similar function l^o is defined for the object.

2.2.5 Temporal Discretization and Solver

The continuous problem is converted into a discrete nonlinear optimization problem using a **direct collocation** approach. The constraints are enforced on the collocation nodes of a time grid matching the video frames.

The resulting sparse nonlinear constrained problem is solved using the **Ceres** solver. The authors utilize the **Pinocchio** library for efficient computation of rigid body dynamics and their analytical derivatives.

3 Main Results of the Paper

3.1 Evaluation Metrics

To evaluate the reconstructed human motion, object motion, and physical quantities, the paper relies on three metrics.

MPJPE (3D Joint Error). The accuracy of the human pose reconstruction is measured by the mean per-joint position error:

$$\text{MPJPE} = \frac{1}{N}\sum_{j=1}^{N} \| \hat{p}_j - p_j^* \|_2,$$

computed after rigid alignment between estimated and ground-truth skeletons.

Force and Moment Errors. For each contact point, the estimated spatial wrench $\hat{w} = (\hat{f}, \hat{m})$ is compared to the force-plate ground truth $w^* = (f^*, m^*)$:

$$E_f = \|\hat{f} - f^*\|_2, \qquad E_m = \|\hat{m} - m^*\|_2.$$

Errors are reported separately for forces and torques.

2D Endpoint Accuracy. For the Handtool dataset, the tool endpoints are manually annotated in 2D. Accuracy is the percentage of frames where the projected 3D endpoints lie within a pixel threshold τ:

$$A(\tau) = \frac{\#\{\|\hat{u} - u^*\|_2 \le \tau\}}{\#\text{annotated frames}}.$$

3.2 Results

The results from the paper were on the Parkour dataset and on the Handtool dataset. The other models used to compare the performances of the paper's model are SMPLify [Bogo et al., 2016] and the HMR 3D human pose estimator [Kanazawa et al., 2018].

Parkour dataset (3D motion). Table 1 shows that incorporating contact constraints and dynamics reduces the 3D joint error compared to SMPLify and HMR across all actions. The improvement is largest for highly dynamic motions (jump, move-up).

Method	Jump	Move-up	Pull-up	Hop
SMPLify	121.75	147.41	120.48	169.36
HMR	111.36	140.16	132.44	149.64
ESTMF	**98.42**	**125.21**	**119.92**	**138.45**

Table 1: MPJPE (mm) on the Parkour dataset. **ESTMF** denotes the method proposed in the original paper.

Parkour dataset (forces). Table 2 reports the average force and torque errors. Errors at the soles are relatively low; hand torques are harder to estimate due to rapid load transfer and high moments during arm-support phases.

Contact	Force	Moment
L. Sole	144.23	23.71
R. Sole	138.21	22.32
L. Hand	107.91	131.13
R. Hand	113.42	134.21

Table 2: Force and moment errors on Parkour (N and N·m).

Handtool dataset (3D motion). Table 3 shows moderate but consistent gains over SMPLify and HMR. Although manual annotations are sparse and noisy, the model benefits from enforcing tool–body consistency over time.

Method	Barbell	Spade	Hammer	Scythe
SMPLify	130.69	135.03	93.43	112.93
HMR	105.04	97.18	96.34	115.42
ESTMF	**104.23**	**95.21**	**95.87**	**114.22**

Table 3: MPJPE (mm) on the Handtool dataset. **ESTMF** denotes the method proposed in the original paper.

Handtool dataset (object endpoints). Table 4 shows that incorporating physical constraints stabilizes object motion and clearly improves endpoint accuracy compared to Mask R-CNN initialization.

Tool	25px	50px	100px
Barbell	38	71	98
Spade	57	86	99
Hammer	61	91	99
Scythe	69	88	98

Table 4: 2D endpoint accuracy (%) for thresholds 25/50/100 pixels.

Illustration. Figure 2 provides representative frames showing input images alongside reconstructed body pose, tool trajectory, and estimated contact forces.

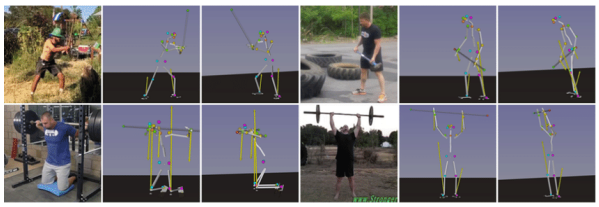

Figure 2: Examples of reconstructed human and object motion on the Parkour and Hand-tool datasets.

4 Limitations of the model

The method, while effective at reconstructing physically consistent human-object interactions, presents several notable limitations. First, the estimation of contact forces and torques remains uncertain: small inaccuracies in pose or object alignment can propagate through the physics model and lead to significant deviations in the recovered wrenches. This issue is amplified by the high computational complexity of the trajectory optimization stage, which restricts the method to offline processing and limits its scalability. Moreover, the performance of the system can vary depending on the mass of the manipulated tool and the physical characteristics of the subject, since these factors influence the underlying dynamics and thus the inferred forces. Finally, the approach inherits the weaknesses of the vision pipeline used in the recognition stage; errors in 2D keypoints, object masks, or contact classifiers directly affect the physical optimization and may produce inconsistent motion estimates. Together, these limitations highlight the sensitivity of the pipeline to both perceptual noise and modelling assumptions.

5 Modernized recognition phase

Since the paper was published in 2019, all of its components from the Recognition Phase are now relatively outdated. Therefore, we propose an improvement of this phase by integrating newer and more recent models.

Human 2D joints While the original paper uses OpenPose, we chose to use one of the state-of-the-art pose estimation models, YOLOv11-pose. Since YOLOv11-pose models are available in different sizes, we selected the YOLOv11-m model, which is both lightweight and accurate, with a number of parameters comparable to the model used in the paper.

Model	$\mathbf{mAP}_{50:95}$	$\mathbf{mAP}_{50}$	CPU	GPU
YOLOv11	**64.9**	**89.4**	**187.3**	**4.9**
OpenPose	61.8	84.9	∼3300	∼45

Table 5: Pose estimation comparison on COCO. The CPU and GPU columns indicate the inference time (in ms).

In Table 5, we observe that the YOLOv11 model outperforms OpenPose in all aspects, both in terms of accuracy and efficiency. Since the models are run on our own computer with limited computational capacity, it is clear that YOLOv11 is a more suitable choice.

Object 2D endpoints For this step, the original paper uses Mask R-CNN combined with PCA to segment and extract tools from the video. We instead explored the YOLOv11 segmentation model, also combined with PCA. The main objective here was to test and validate the full pipeline on other types of videos, such as a human using a baseball bat.

Contact states $\delta_j(t)$ In this part, the paper uses a ResNet classifier applied to joint image patches. We kept this model unchanged, as it is simple to use and already provides good performance.

3D Pose Estimation Finally, the original paper proposes the Human Mesh and Pose Recovery (HMR) model [Kanazawa et al., 2018], which performs 3D pose estimation from a single frame. We explored the more recent VIBE model [Kocabas et al., 2020], which is able to perform 3D estimation from a single RGB frame and can also leverage temporal information from videos.

The results shown in Table 6 are taken from the official VIBE paper. We can see that this model significantly outperforms HMR across all evaluation metrics.

	3DPW				3DHP			H36M	
Model	PA	MPJPE	PVE	Accel	PA	MPJPE	PCK	PA	MPJPE
HMR	76.7	130.0	–	37.4	89.8	124.2	72.9	56.8	88.0
VIBE	**51.9**	**82.9**	**99.1**	**23.4**	**64.6**	**96.6**	**89.3**	**41.4**	**65.6**

Table 6: Comparison of HMR and VIBE on 3DPW, MPI-INF-3DHP, and Human3.6M. Lower values are better for PA, MPJPE, PVE, and Accel, while higher values are better for PCK.

6 New pipeline results

Using these updated models, we were able to reproduce the full pipeline. We compared the results obtained with the original pipeline (using the demo from the official GitHub implementation) to those produced by our new pipeline. Both pipelines functioned correctly; however, we observed some small differences in the results. In the original vision processing pipeline, occasional errors appeared, whereas our pipeline was almost perfectly accurate on the tested video. For instance, Figure 3 shows an error in the top-left corner of the image that leads to a shift in the body pose estimation, which is no longer properly centered.

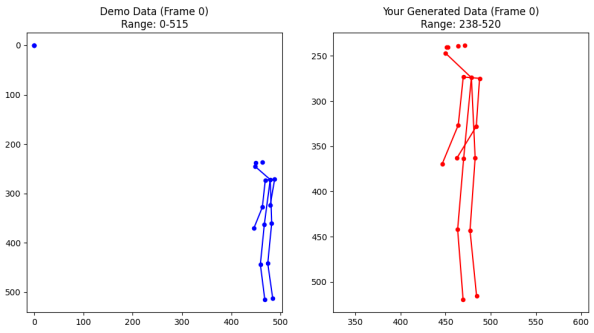

Figure 3: Comparison between the original vision pipeline (left), showing a false positive detection in the top-left corner, and our updated pipeline (right), without this error.

Additionally, using the same small demo to compare both approaches, we observed a slight improvement in convergence with our new method compared to the original one. This improvement is likely linked to the reduced number of errors produced by the vision pipeline.

Finally, we ran our new pipeline on additional videos to evaluate its generalization and efficiency. Figure 4 presents results obtained on new sequences, including a video of a man performing jumping jacks and another of a man swinging a baseball bat. In both cases, the results were very satisfactory.

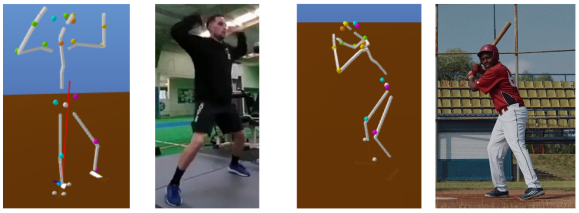

Figure 4: Example results obtained with the new pipeline on previously unseen videos.

Due to the limited time available for this project and our constrained computational resources, we were not able to perform a full quantitative evaluation on a large dataset to precisely assess the accuracy improvements. However, the gain in efficiency, particularly in terms of runtime, appears to be significant and worth highlighting.

7 Improvements ideas and limitations

Even though the paper already offers a great tool and even though we tried to tackle some of its limitations with our new recognition implementation, some problems remain :

- **Estimation of Forces :** The force estimation still lacks some accuracy. This problem is due to the fact that this pipeline has no data on the mass of the people in the video, or on the mass of the object. Thus having an accurate force estimation becomes way harder. A solution to this problem would be to train another model that would only focus on estimating the mass of the objects in the video.
- **Computational complexity :** Even though we proposed a faster and lighter recognition phase, the overall computational complexity stays very high. This significantly limits real-time deployment and constrains scalability to longer video sequences or multiple subjects.

- **Decoupled 2D/3D pose estimation:** In the current workflow, 2D keypoints (OpenPose/YOLOv11) and 3D poses (HMR/VIBE) are inferred by separate models without enforcing mutual consistency. As a result, the optimizer may receive contradictory cues: a 3D skeleton that does not accurately reproject onto the detected 2D joints. This lack of cross-level coupling can introduce instability and additional motion jitter.
- **Difficulty in weighting loss terms:** Selecting appropriate coefficients for the different components of the objective function remains largely empirical. Achieving the correct trade-off between physical plausibility, measurement fidelity, and temporal smoothing requires extensive manual tuning. This sensitivity to hyperparameters limits scalability to new datasets and suggests the need for automatic or adaptive weighting strategies.
- **Uncertainty propagation through the pipeline:** Errors originating from the vision stage (2D keypoint noise, segmentation drift...) propagate through the physical optimizer and are amplified when estimating velocities, accelerations, and torques. The current formulation provides point estimates without uncertainty bounds, which limits the interpretability of the recovered forces. A possible improvement would be to incorporate uncertainty-aware optimization or Bayesian pose estimation to quantify confidence and improve robustness.
- **Simplified contact modelling:** Contact states are treated as binary variables and friction cones are approximated coarsely. These simplifying assumptions can lead to unrealistic contact transitions or slipping artifacts. More expressive contact models (like soft constraints or data-driven friction estimation) would improve contact realism and force estimation.

8 Conclusion

In this report, we reviewed the method proposed in Estimating 3D Motion and Forces of Person–Object Interactions from Monocular Video, highlighting its originality in unifying visual perception with physics-based trajectory optimization. The authors demonstrate that accurate and physically consistent reconstructions of human–object interactions can be achieved using only monocular RGB video, a significant step toward scalable motion understanding without laboratory equipment. Despite its strong performance, the method remains limited by perceptual noise, high computational cost, and sensitivity to modelling assumptions such as object mass and contact states.

We then proposed a modernized recognition pipeline that replaces outdated components with state-of-the-art models including YOLOv11 for 2D keypoint detection, improved segmentation via YOLOv11-Seg, and VIBE for 3D pose estimation. These upgrades yield more stable observations, faster computation, and overall cleaner inputs for the physical optimizer. Although a full quantitative evaluation on large datasets was infeasible due to time and resource constraints, qualitative comparisons and small-scale experiments indicate more robust vision outputs and improved convergence behaviour.

Our work shows that updating the recognition stage alone can substantially enhance the practicality of physics-based motion reconstruction systems. Future progress will likely require better mass and force estimation, hybrid 2D–3D learning frameworks, and more efficient solvers to reduce computational overhead. Ultimately, bridging modern deep-learning perception with principled physical modelling remains a promising direction toward accurate understanding of human–object interactions in unconstrained environments.

References

[Bogo et al., 2016] Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., and Black, M. J. (2016). Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[Kanazawa et al., 2018] Kanazawa, A., Black, M. J., Jacobs, D. W., and Malik, J. (2018). End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Recognition (CVPR)*.

[Kocabas et al., 2020] Kocabas, M., Athanasiou, N., and Black, M. J. (2020). Vibe: Video inference for human body pose and shape estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Li et al., 2019] Li, Z., Sedlar, J., Carpentier, J., Laptev, I., Mansard, N., and Sivic, J. (2019). Estimating 3d motion and forces of person-object interactions from monocular video. In *Computer Vision and Pattern Recognition (CVPR)*.