JuggleRL

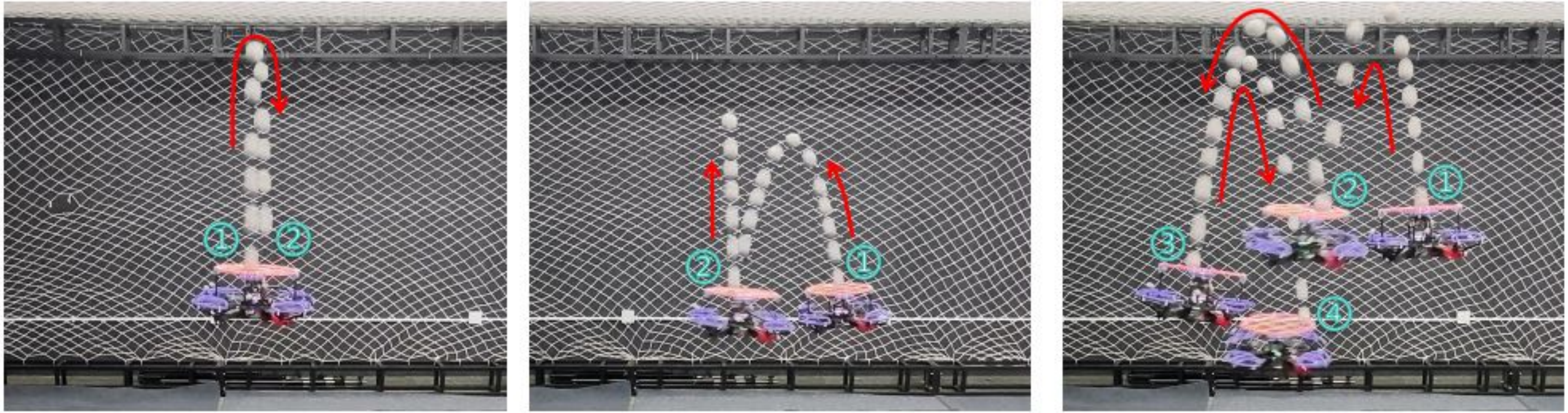
Mastering Ball Juggling with a Quadrotor via Deep Reinforcement Learning

Background

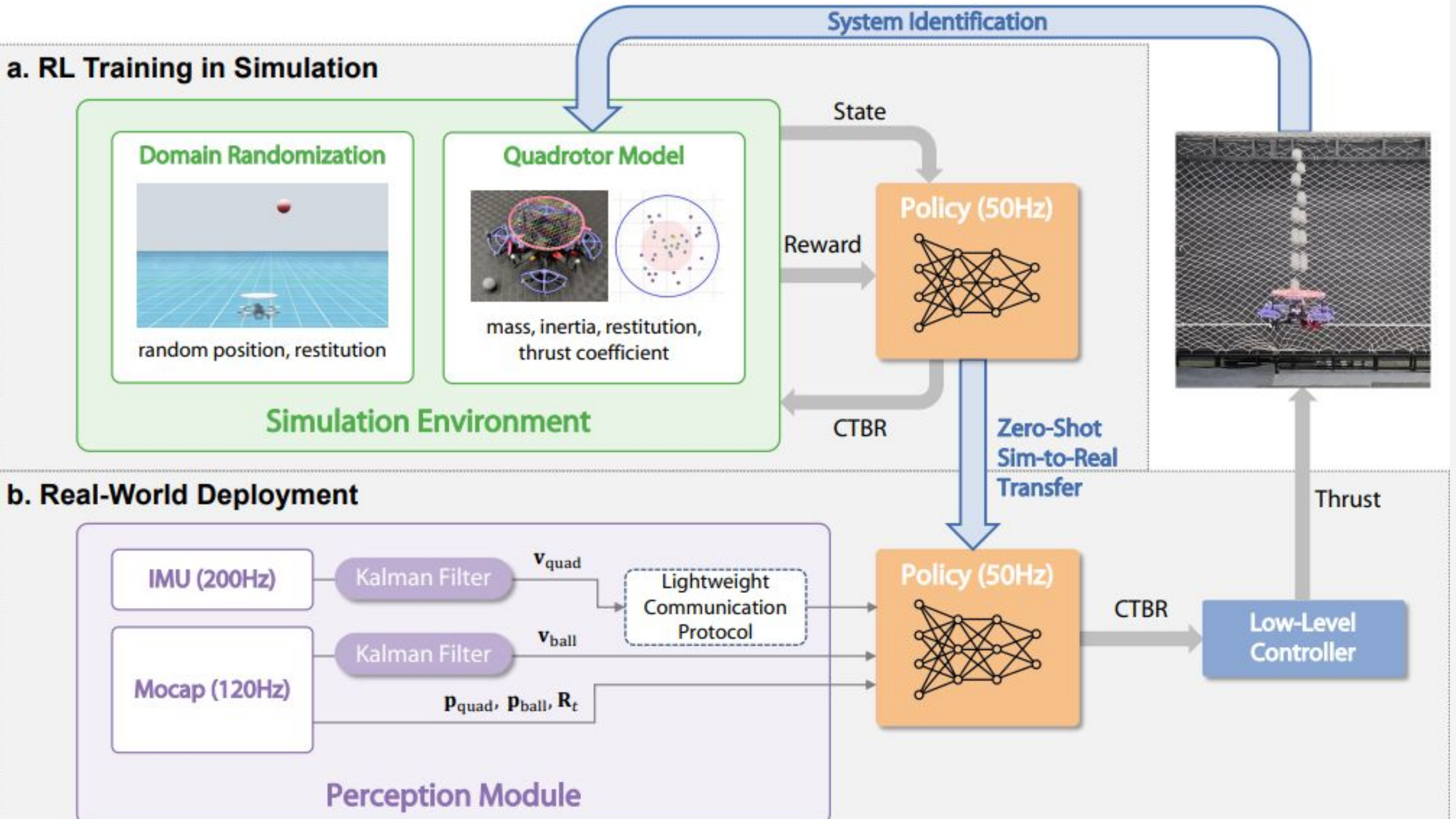
- Dynamic interaction with the environment remains one of the central challenges in robotics
  - Robots have to deal with noise, model uncertainty and environmental variability
  - Aerial robots are especially difficult as they must maintain stability all while making precise adjustments based on their chosen interaction
- Challenge: Train a quadrotor to ‘juggle’ a ball in the air
  - Originally addressed this problem by predicting ball trajectories, calculating contact points, and planning quadrotor motions to satisfy these constraints
  - Limited by nonlinear ball dynamics, error accumulation, and hard trajectory constraints

Innovations

- Implemented Deep Reinforcement Learning strategies to calculate the drone's trajectories over many episodes of training
  - Able to adapt to error accumulation and break trajectory restraints from traditional methods
- "JuggleRL achieves up to 462 hits with an average of 311 in real experiments under varying ball drop heights, far exceeding the baseline, which reaches at most 14 hits with an average of 3.1."



(a) Juggling near the origin (b) Directly correcting small deviations (c) Two-stage "stabilize-then-correct" strategy



System Identification
a. RL Training in Simulation
Domain Randomization
random position, restitution
Quadrotor Model
mass, inertia, restitution,
thrust coefficient
Simulation Environment
State
Reward
CTBR
Policy (50Hz)
Zero-Shot
Sim-to-Real
Transfer
b. Real-World Deployment
IMU (200Hz)
Kalman Filter
$\mathbf{v}_{\text{quad}}$
Lightweight
Communication
Protocol
Mocap (120Hz)
Kalman Filter
$\mathbf{v}_{\text{ball}}$
$\mathbf{p}_{\text{quad}}$, $\mathbf{p}_{\text{ball}}$, $\mathbf{R}_t$
Perception Module
Policy (50Hz)
CTBR
Low-Level
Controller
Thrust

Conclusion

- Successfully demonstrated the first reinforcement learning-based system for continuous aerial ball juggling with a quadrotor.
- The trained policy achieved insane performance in real-world experiments, sustaining an average of 311 consecutive hits (with a maximum of 462).

- Proved that deep RL can solve highly complex, contact-rich robotic tasks that are extremely challenging for traditional methods.
- Successful Sim-to-Real Transfer: Showcased a "zero-shot" sim-to-real transfer by combining large-scale simulation with careful system identification (calibrating drone and ball physics) and domain randomization.

- Limitations: All data was gathered through external cameras and videos
- Future: Maybe have all of the sensors onboard the drone?

Thank you

Source
https://arxiv.org/pdf/2509.24892