7th INTERNATIONAL CONFERENCE OF THE ROBOTICS SOCIETY

AIR 2025 Advances in Robotics

Targeted Object Striking for a 7-DoF Manipulator: A Residual Learning Approach

Priyansh Sinha[1], Rishin Chakraborty[1], Samarth Brahmbhatt[2], Nagamanikandan Govindan[1]
[1]International Institute of Information Technology, Hyderabad [2]Independent Researcher

1. Introduction & Motivation

While grasping is the most common form of robotic manipulation, non-prehensile actions like striking offer unique advantages in speed and reach. By striking an object, a robot can move it far beyond its own physical arm length, enabling powerful applications in logistics (rapidly sorting packages), manufacturing (clearing workspaces), and even service robotics (clearing a table).

The Problem: How can a robot strike an object so that it slides and stops precisely at a desired target location?

The Challenge: Modelling High-Velocity Impact Dynamics
The core difficulty lies in the complex, high-speed physics of the impact. Creating a perfect mathematical model is almost impossible.

- **Unmodeled Physics**: Real-world factors like non-uniform surface friction, the exact geometry of the contact point, material deformation, and even minor deviations due to air resistance are incredibly difficult to model accurately.
- **Simulation Parameter Tuning**: Manually tuning a simulator's parameters (a process called system identification) is a fragile solution. There are a limited number of parameters that any simulator accounts for and it can be very difficult to get an accurate simulation by tuning these.
- **Consequence**: Actions planned in a flawed simulation result in significant errors in the real world, making the robot unreliable.
- **Our Solution:** We propose a hybrid framework that combines a physics-based simulator with a data-driven machine learning model. Our framework accepts that the simulator will be imperfect. Instead of trying to fix the simulation, we use a data-driven approach to learn a model of the simulation's errors (a residual model) and use it to correct the robot's actions in real-time.

2. Methodology: A Hybrid Approach

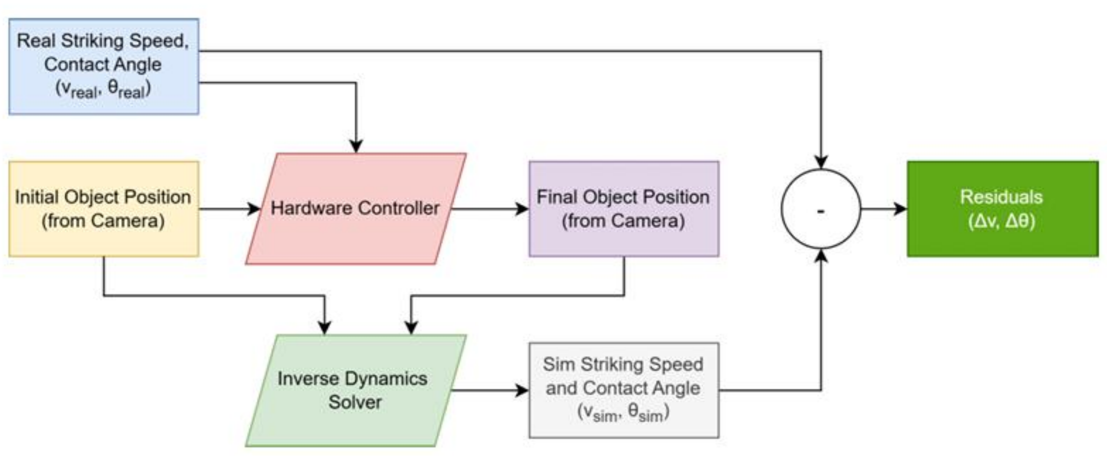
Our goal is to find the correct striking **speed (v)** and **angle (θ)** to send an object to its target. We do this in three steps:

Step 1: Inverse Dynamics Solver (in Simulation)

- We use a MuJoCo physics simulation and an NLopt optimizer to find the *ideal* striking parameters (v_{sim}, θ_{sim}) that would work in a "perfect" simulated world.
- This gives us a good first guess, but it's not accurate enough for the real world.

Step 2: Residual Learning (Bridging the Sim-to-Real Gap) This is the core of our contribution. We model the difference between the simulator's prediction and the required real-world action as a **residual**.

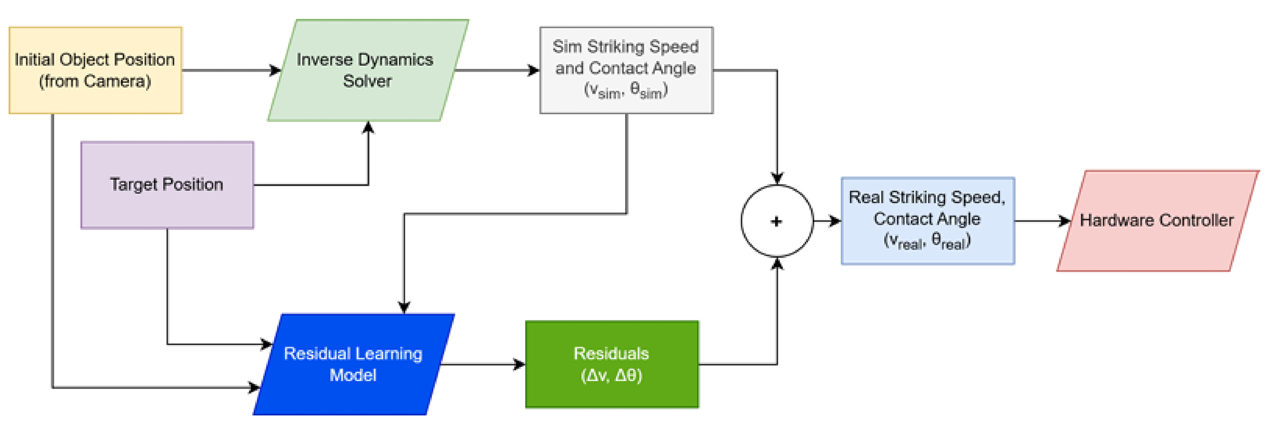
- We collect data by performing random strikes in the real world.
- For each real strike (v_{real}, θ_{real}), we calculate what the simulator *would have* predicted as the command parameters to achieve this strike in the simulation (v_{sim}, θ_{sim}).
- The error, or residual, is calculated:
  - $\Delta v = v_{real} - v_{sim}$
  - $\Delta\theta = \theta_{real} - \theta_{sim}$
- We train a simple Neural Network (MLP) to predict these residuals based on the target path and the simulator's output.



Residual Learning Pipeline

Step 3: Execution on the Robot To strike an object to a new target:

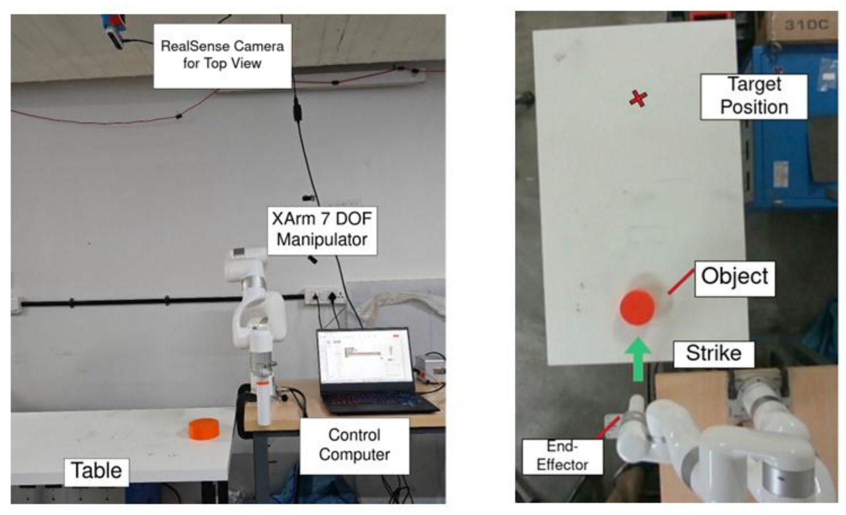
1. The **Inverse Dynamics Solver** calculates the ideal v_{sim} and θ_{sim}.
2. The trained **Residual Model** predicts the necessary correction, Δv and $\Delta\theta$.
3. The final command sent to the robot is the sum of both:
   - $v_{real} = v_{sim} + \Delta v$
   - $\theta_{real} = \theta_{sim} + \Delta\theta$



Execution Pipeline

3. Results & Key Findings

We compared our Residual Learning approach against a baseline System Identification method, where we only tuned the simulator's physics parameters. The expirements were performed using the uFactory xArm 7, a 7-DOF Manipulator, equipped with a cylindrical striking end-effector.



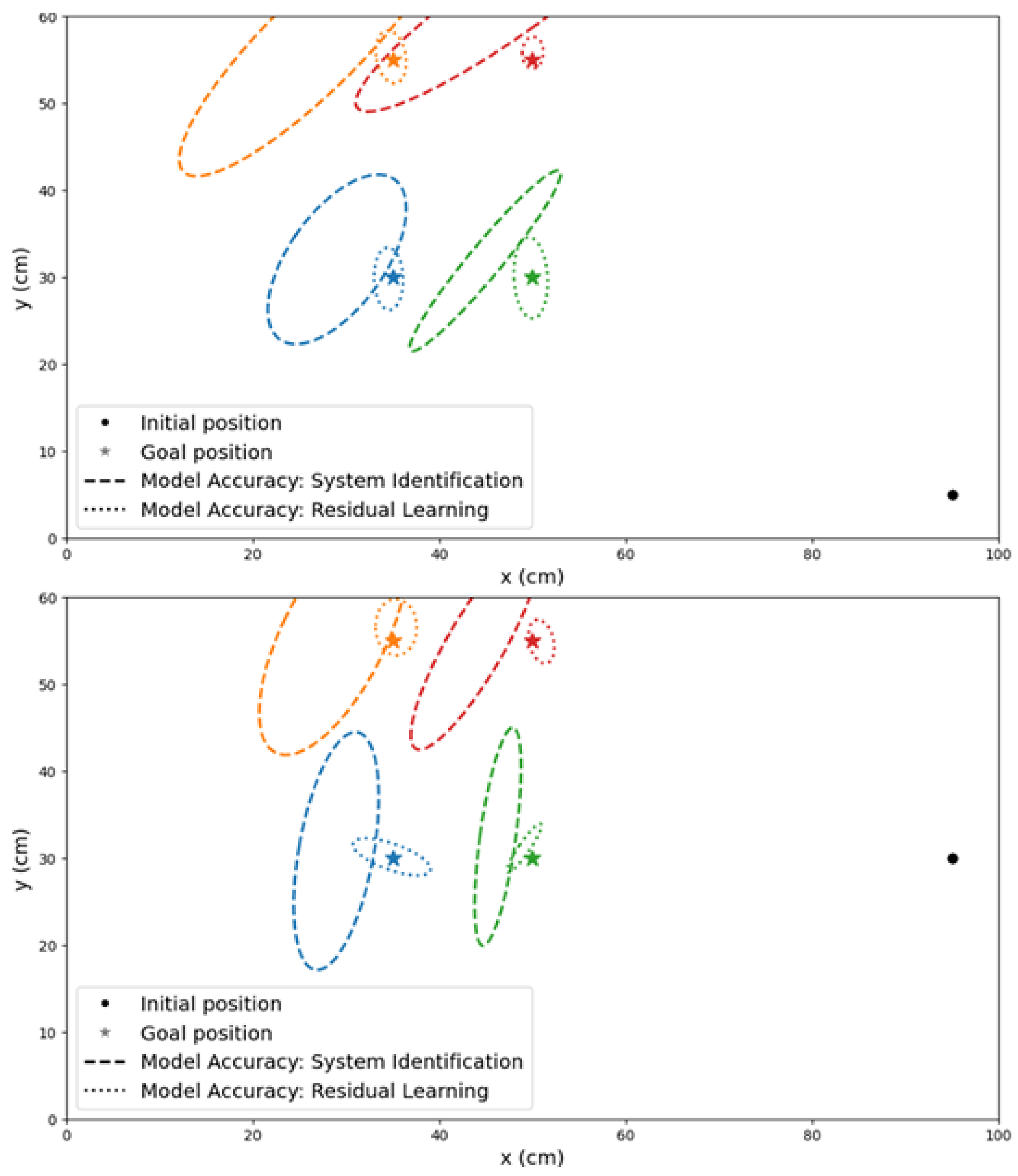
(a) Side view (b) Top view

Experimental Setup

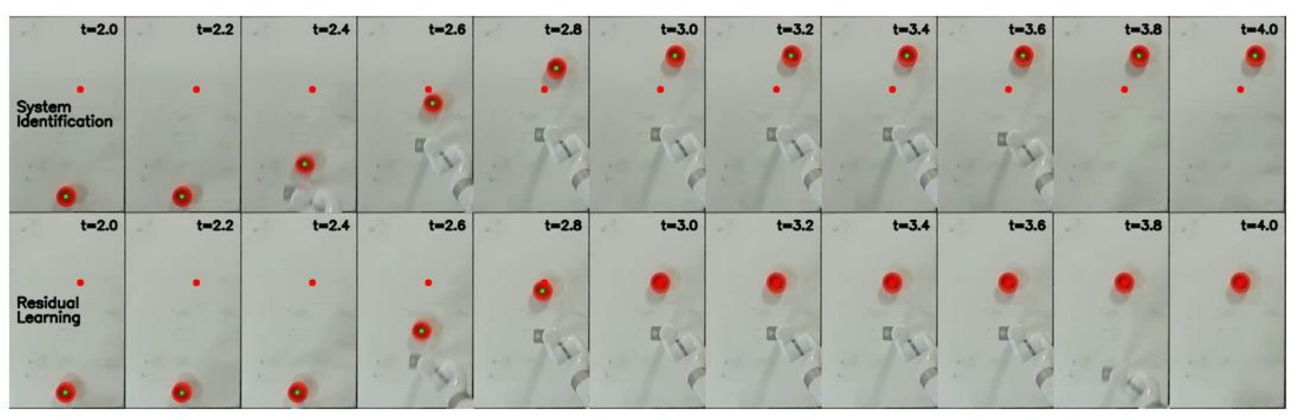
Finding 1: Residual Learning is Significantly More Accurate Our proposed method dramatically reduces the error between the object's final position and the target.

- **System Identification Error:** 9.12 cm (average)
- **Residual Learning Error:** 1.68 cm (average)

This represents an **81.6% reduction in error**, demonstrating the effectiveness of learning to compensate for unmodeled dynamics.



Comparison between System Identification (dotted lines) and our Residual Learning Approach (solid lines)



Comparison between System Identification (top) and our Residual Learning Approach (bottom)

Finding 2: The Approach Generalizes We validated our method on a second object (a cube) with different physical properties (mass, size, friction) and achieved similarly high accuracy, reducing the mean error from 10.02 cm to 4.36 cm.

4. Conclusion & Future Work

Conclusion We introduced a novel hybrid framework for robotic striking that effectively bridges the simulation-to-reality gap. By combining a physics-based solver with a learned residual model, our system can strike objects to a target location with high accuracy and precision. This approach is more effective than traditional system identification because it can compensate for all sources of sim-to-real mismatch, both known and unknown.

Future Work

- **Generalization:** Develop a single model that can adapt to objects of any shape and size without needing to be retrained.
- **Dynamic Adaptation:** Create a framework that can explore a new environment at runtime and tune its residuals on-the-fly.

Priyansh Sinha: priyansh.sinha@research.iiit.ac.in
Rishin Chakraborty: rishin.chakraborty@research.iiit.ac.in

03-07-2025