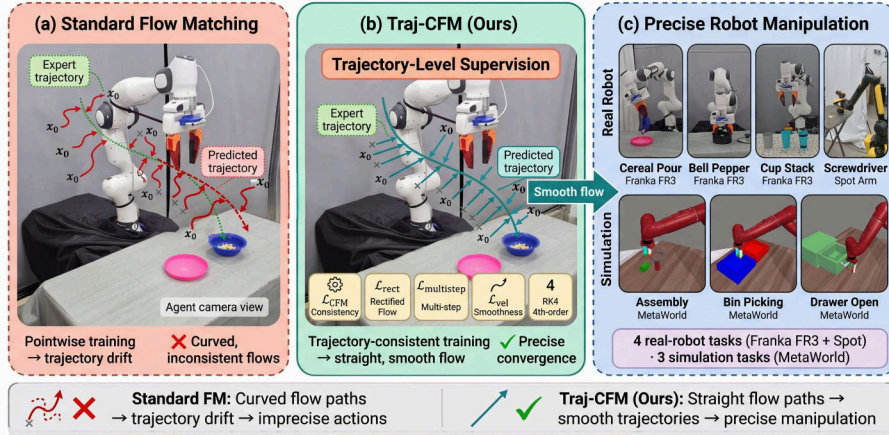


Overview



Standard flow matching trains pointwise velocity targets, producing curved flow paths that drift under integration (a). Traj-Consistent FM (Traj-CFM) enforces trajectory-level supervision through four training losses combined with RK4 inference, yielding straight, consistent paths (b), enabling precise manipulation across four real-robot tasks and three simulated MetaWorld tasks (c).

70%

Bell Pepper Placing overall – baselines 0%

60%

Cup Stacking overall – baselines 0%

100%

Screwdriver Placement, precision tool-tip

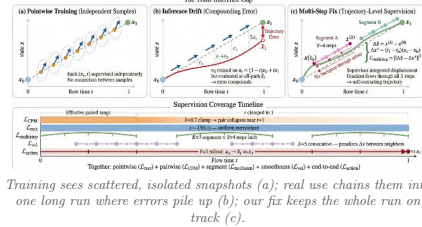
6,750×

lower discretization error, RK4 vs. Euler

1. Motivation: The Train – Inference Gap

A robot policy learns by looking at scattered, one-off snapshots of motion. But to actually do a task, it has to chain hundreds of these small predictions together, back to back, in one continuous run. That mismatch – practiced in isolated pieces, but used as one long connected sequence – is the root problem, and it breaks in three ways:

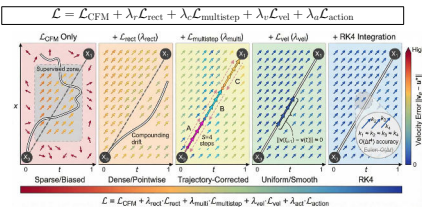
- **It never rehearses the middle of the motion.** Training mostly pairs the start and the end; the in-between path is barely supervised.
- **Small mistakes snowball.** Like a long road trip with no GPS correction, tiny errors at each step add up the longer the robot runs – until the whole action misses.
- **Predictions can get jittery.** With no smoothness signal, one step's output can swing sharply from the last, causing shaky, unreliable motion.



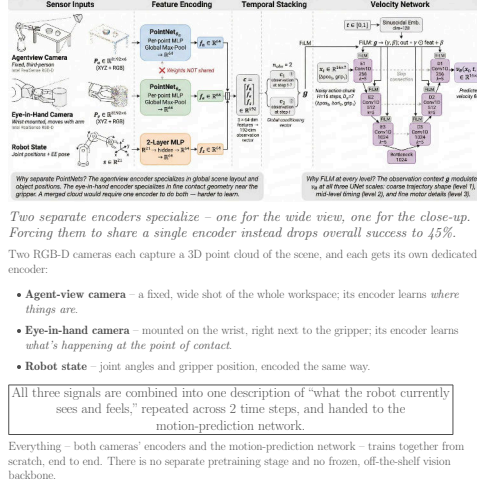
2. Method: Four Components, Jointly

Traj-Consistent FM closes the gap with four complementary components:

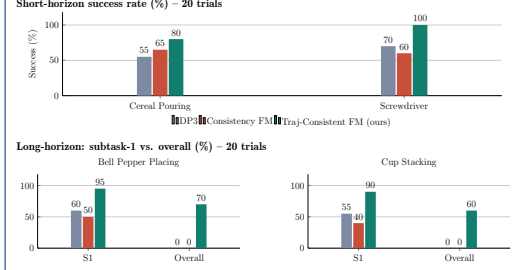
1. **Rectified velocity regression** – dense supervision across $t \in [0, 1]$, not just clamped schedule pairs.
 $\mathcal{L}_{\text{rect}} = \mathbb{E} \|v_t(x_t, t, c) - (x_1 - x_0)\|^2$
2. **Multi-step trajectory consistency** – supervises the integrated $S=4$ -step displacement directly, closing the gap at its source.
 $\mathcal{L}_{\text{multistep}} = \mathbb{E} \|\Delta x - \Delta x^*\|^2$
3. **Velocity smoothness regularization** – penalizes oscillation between neighboring times, so higher-order solvers stay accurate.
 $\mathcal{L}_{\text{vel}} = \mathbb{E} \left\| \frac{1}{S-1} \sum_{t=1}^{S-1} \|v_t(t_{i+1}) - v_t(t_i)\|^2 \right\|$
4. **RK4 inference** – 4th-order solver: $\mathcal{O}(\Delta t^4)$ global error vs. $\mathcal{O}(\Delta t)$ for Euler; 30 steps, 120 network evaluations.



3. Method: Dual-View Point Cloud Encoder



5. Results: Real Robot & Simulation



Baselines learn competent single-phase policies (40–60% S1) but never complete a full multi-phase trial (0% overall) – pointwise-trained fields drift too far across extended integration.

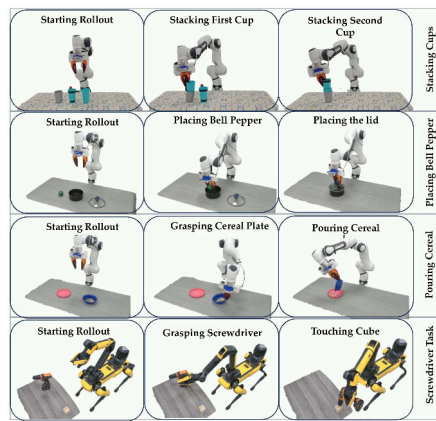
Traj-Consistent FM makes zero cup-discrimination errors across all 20 Cup Stacking trials – the two cups differ only by a thin stripe, and $\mathcal{L}_{\text{multistep}}$'s S -step gradient amplification is what forces the encoder to resolve it.

MetaWorld simulation (50 episodes) – reward by task

Task	DP3	Cons.	FM	Ours
Drawer Open	376.3	386.6	446.3	
Assembly	241.6	293.3	281.2	
Bin Picking	88.3	88.5	89.5	

Success rates are near-saturated for all methods (96–100%); reward – which penalizes inefficient or jerky motion – separates them: +15% on Drawer Open.

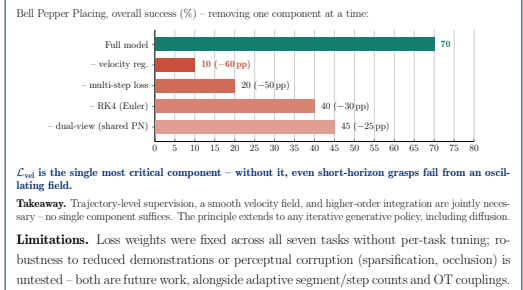
4. Setup: Real-Robot & Simulation Tasks



4 real-robot tasks (Franka FR3 + Boston Dynamics Spot) + 3 MetaWorld simulation tasks.

Baselines. DP3 – Ze et al., 3D Diffusion Policy, RSS 2024. Consistency FM – Zhang et al., FlowPolicy, AAI 2025.

6. Ablation & Takeaways



7. Conclusion

- **Solves the failure that matters.** Where strong published baselines score 0% on long-horizon manipulation, Traj-Consistent FM reaches 70–100%.
- **Generalizes beyond flow matching.** Trajectory-level correction of pointwise-trained velocity fields applies to diffusion policies too.
- **Zero tuning overhead.** One fixed loss configuration transfers across seven tasks and two robot platforms, no per-task sweep.

Traj-Consistent FM turns flow-matching policies into long-horizon, real-robot-ready controllers.