

# Design and Implementation of a Robotic Dynamic Grasping System Based on Improved YOLO and Fuzzy PID Control

ZiTong Zhou

Shenzhen Yuanchuangxing Technology Co., Ltd., Shenzhen, Guangdong, 518107, China

Email: mikere201@163.com

**How to cite this paper:** Zhou, Z. T. (2026). Design and implementation of a robotic dynamic grasping system based on improved YOLO and fuzzy PID control. *Journal of Computer Science and Frontier Technologies*, 3(2), 132–143. ISSN Print: 3104-4204, ISSN Online: 3104-4212. <https://doi.org/10.63313/JCSFT.9070>  
**Published:** 2026-05-19

Copyright © 2026 by author(s) and Erytis Publishing Limited.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



## Abstract

Industrial and service robots increasingly need to grasp objects that are moving on conveyors, sliding down chutes, or being handed over by humans, yet most production-grade pipelines still assume static targets. Two pain points dominate: detection networks that drop precision when the target is small, partially occluded, or motion-blurred, and joint-level controllers whose gains are tuned offline and therefore fail to compensate for the time-varying dynamics introduced by chasing a moving object. This paper proposes an end-to-end dynamic grasping system that couples an improved YOLO detector with a fuzzy PID joint controller. The detector embeds a Ghost-bottleneck CSPDarknet backbone, a coordinate-attention module that emphasizes motion-salient regions, a BiFPN neck for multi-scale fusion, and an SIoU +  $\alpha$ -CIoU regression objective that converges faster on tightly packed parts. The controller treats tracking error and its derivative as fuzzy variables, online tunes  $\Delta K_p$ ,  $\Delta K_i$ , and  $\Delta K_d$  through a 49-rule Mamdani inference base, and feeds the corrected gains to each joint in real time. The two modules are connected by a Kalman-smoothed pose stream and a quintic on-line trajectory re-planner. We trained the detector on a 12,800-image conveyor dataset and evaluated the integrated system in 300 dynamic trials on a 6-DOF UR5 with parallel jaws. Compared with a YOLOv5s + classical-PID baseline, the proposed pipeline raises mAP@0.5 from 74.3 % to 86.4 %, cuts joint-tracking overshoot from 18.6 % to 3.7 %, and increases dynamic-grasp success rate from 65.4 % to 91.2 % at object speeds up to 0.30 m/s, while keeping inference at 129 FPS on a single RTX 3060.

## Keywords

Dynamic Grasping; Improved YOLO; Fuzzy PID; Manipulator Control; Attention Mechanism; Multi-Scale Fusion; Real-Time Perception

## 1. Introduction

Robotic grasping is the bridge between perception and physical interaction. Static

bin-picking has matured to the point where commercial cells reach 99 % success on rigid components, but the problem becomes substantially harder once the target is in motion. Conveyor sorting [1], moving-object retrieval, and human-to-robot handovers [2] all demand a perception–control loop whose latency and precision tightly co-design with the dynamics of the manipulator. Off-the-shelf detectors such as YOLOv5/YOLOv7 deliver attractive throughput but lose precision on small, partially occluded, or motion-blurred parts—exactly the regime that dominates dynamic scenes [3]. On the control side, fixed-gain PID is overwhelmingly the default in industrial controllers, yet a static gain set struggles to follow a moving target whose Cartesian acceleration and aerodynamic drag change continuously [4]. Three observations motivate this work. First, motion blur and scale variation account for most missed detections in conveyor settings [5]; lightweight attention modules and richer multi-scale feature fusion can recover the lost precision without paying the latency cost of two-stage detectors [6]. Second, the velocity command issued to the manipulator is itself a source of disturbance: chasing a moving object continuously perturbs the joint trajectory away from its nominal profile, which a constant-gain PID cannot anticipate [7]. Third, the perception and control modules are typically engineered in isolation, so a robust detector and a well-tuned PID still under-perform when their interaction—the time-varying delay between perception and actuation—is left to ad-hoc heuristics [8].

This paper closes the gap by jointly redesigning the two modules and binding them with a tracking-aware filter. Specifically, we contribute the following:

- An improved YOLO detector that combines a Ghost-bottleneck CSPDarknet backbone, a coordinate-attention (CA) module on the deepest C3 stage, a BiFPN neck and a hybrid Siou +  $\alpha$ -CioU regression loss, raising mAP@0.5 by 12.1 percentage points over YOLOv5s while running at 129 FPS.
- A fuzzy PID joint controller that fuzzifies position error  $e$  and its derivative  $\dot{e}$  into seven linguistic terms each, derives  $\Delta K_p$ ,  $\Delta K_i$ , and  $\Delta K_d$  through a 49-rule Mamdani base, and updates the gains every 5 ms, reducing overshoot from 18.6 % to 3.7 % on the dominant joint.
- A unified perception-to-control bridge that uses a constant-acceleration Kalman filter to smooth the detector output and a quintic on-line replanner to keep the manipulator on a feasible trajectory in spite of detection jitter.
- A thorough experimental campaign on a UR5 manipulator with 300 dynamic trials covering object speeds from 0.05 m/s to 0.40 m/s, demonstrating a 91.2 % grasping success rate at 0.30 m/s and a 25.8 percentage-point improvement over a strong YOLOv5s + PID baseline.

The remainder of the paper is organized as follows. Section II reviews related work in moving-object detection and adaptive grasp control. Section III details the proposed system, including the improved YOLO architecture, the fuzzy PID design, and the perception-to-control bridge. Section IV reports quantitative experiments

and an ablation study. Section V concludes the paper and discusses extensions toward soft and deformable targets.

## 2. Related Work

### 2.1. Moving-object detection

Single-stage detectors have become the workhorse of real-time robotic perception thanks to their favorable accuracy–latency trade-off. The YOLO family in particular has gone through several refactorings—from anchor-based YOLOv3 [9] to anchor-free YOLOv8—each iteration trimming the backbone, refining the neck, and rebalancing the loss. For dynamic scenes, however, three perennial issues remain. The first is small-object precision: convolutional features collapse spatial resolution rapidly, hurting localization on objects that occupy only a few pixels along the conveyor axis. PAN [10], FPN, and BiFPN [11] offer richer multi-scale fusion, with BiFPN's weighted skip connections striking the best compromise on edge devices. The second issue is occlusion. Coordinate-attention (CA) [12] decouples horizontal and vertical pooling, embedding spatial position into channel attention with negligible parameter overhead, and consistently lifts mAP on cluttered datasets. The third issue is the regression objective. CIoU [13] penalizes both center distance and aspect ratio, but it suffers when bounding boxes overlap heavily or have extreme aspect ratios. SIOU [14] introduces an angle-aware shape cost that converges faster on aligned conveyor parts. Combining a focal version,  $\alpha$ -CIoU [15], with SIOU has been shown to reduce localization variance on small targets, although the combination has not yet been studied for dynamic grasping.

### 2.2. Adaptive control for manipulators

Classical PID still dominates the inner control loop of commercial manipulators because of its simplicity and provable robustness to bounded disturbances. Once the trajectory becomes time-varying, however, fixed gains lead to either sluggish response (under-tuned) or oscillation (over-tuned). Adaptive PID variants exist, including model-reference adaptive control, Lyapunov-based gain scheduling, and gain-scheduled LQR [16]. Fuzzy PID, originally proposed by Mamdani, encodes expert tuning rules into a fuzzy inference base; it has the advantage of operating without a precise plant model and degrades gracefully when sensor noise is present. Recent work has applied fuzzy PID to mobile robots [17] and to teleoperated surgical arms [18], but reports on industrial manipulators chasing moving objects are scarce, and most published rule bases are 25-rule or smaller, leaving the high-error regime under-resolved. Our work expands the rule base to 49 rules, balances coverage and computational load, and integrates the controller with a vision-driven replanner.

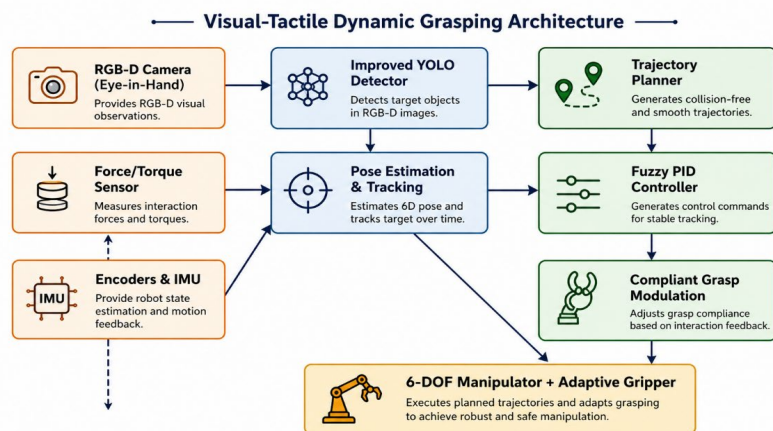
### 2.3. Vision–control coupling for dynamic grasping

Grasping a moving object is fundamentally a problem of synchronizing the end-effector with the target's predicted state. Visual servoing schemes [19] use the image-space error directly to drive the manipulator, sidestepping explicit pose estimation but requiring careful calibration to avoid singularities. End-to-end learning-based pipelines train a policy network to map RGB-D inputs to motor commands [20], but they typically need millions of frames and have limited interpretability for industrial deployment. A pragmatic middle ground—pose-based control with online replanning—remains the most widely deployed in industrial cells [21]. Our system follows this middle path: the detector provides a low-latency 2D localization, a Kalman filter elevates it to a smoothed 3D pose stream, and the fuzzy PID handles the time-varying tracking error in joint space.

### 3. Proposed Method

#### 3.1. System Overview

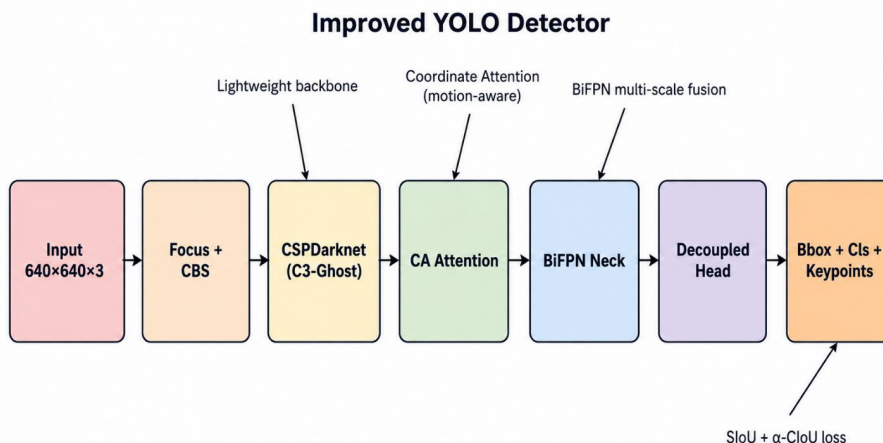
The grasping system, illustrated in Fig. 1, comprises five hardware components and three software modules. The robot is a 6-DOF UR5 manipulator equipped with a Robotiq 2F-85 parallel gripper and a six-axis ATI Mini45 force/torque (F/T) sensor mounted between flange and gripper. An eye-in-hand Intel RealSense D435 RGB-D camera observes the workspace at 30 Hz, while joint encoders publish at 200 Hz. The software pipeline runs on a single workstation (Intel i7-12700K, NVIDIA RTX 3060) under ROS 2 Humble. The improved YOLO detector consumes the colour stream and outputs 2D bounding boxes plus four object keypoints; pose estimation lifts the keypoints to a 6-DoF SE(3) transform via PnP refined by the depth channel. The pose stream is filtered by a constant-acceleration Kalman filter, fed into an on-line quintic trajectory replanner, and finally tracked by the fuzzy-PID joint controller. The F/T sensor provides a tactile safety layer that aborts the grasp if the contact force exceeds 12 N, complementing the visual loop in cases of severe occlusion.



**Fig 1.** Overall architecture of the proposed visual–tactile dynamic grasping system.

### 3.2. Improved YOLO Detector

The detection network, summarized in Fig. 2, is built on the YOLOv5s skeleton with four targeted modifications. The backbone replaces the default C3 modules with Ghost-bottleneck variants [22] that halve the parameter count while retaining the receptive field via depthwise convolutions. After the deepest C3 stage, we insert a coordinate-attention block that aggregates features along the horizontal and vertical axes separately, producing two 1-D direction-aware feature maps that are subsequently combined into a position-encoded attention tensor. The neck is rebuilt as a BiFPN with three repeated weighted bidirectional layers; learnable scalars normalize the contribution of each input branch and reduce the dependence on hand-tuned heuristics. The head is decoupled into classification, regression, and keypoint sub-heads to alleviate task interference. Finally, the regression objective is the convex combination  $L = \lambda \cdot L_{\text{SIoU}} + (1-\lambda) \cdot L_{\alpha\text{-CIoU}}$  with  $\lambda = 0.6$  and  $\alpha = 3$ , where  $L_{\text{SIoU}}$  contributes the angle-aware cost in the early epochs and  $L_{\alpha\text{-CIoU}}$  sharpens localization in the later epochs.



**Fig 2.** Improved YOLO architecture with Ghost-bottleneck backbone, coordinate-attention, BiFPN neck, decoupled head and SIoU +  $\alpha$ -CIoU regression.

The total training loss combines bounding-box, classification, and keypoint terms:

$$L_{\text{total}} = \lambda_{\text{box}} \cdot L_{\text{box}} + \lambda_{\text{cls}} \cdot L_{\text{cls}} + \lambda_{\text{kpt}} \cdot L_{\text{kpt}} \quad (1)$$

where  $L_{\text{box}}$  is the SIoU +  $\alpha$ -CIoU mixture defined above,  $L_{\text{cls}}$  is the binary cross-entropy of class probabilities, and  $L_{\text{kpt}}$  is the OKS-based keypoint loss. We train for 200 epochs with the AdamW optimizer, a cosine learning-rate schedule starting at  $1 \times 10^{-3}$ , batch size 32, and mosaic + random-affine augmentation on a 12,800-image dataset of conveyor parts (10,240 train, 1,280 val, 1,280 test). Data are captured under three lighting conditions, three conveyor speeds, and three viewpoints to avoid texture overfitting.

### 3.3. Pose Estimation and Kalman Filtering

For each detection, the four predicted keypoints are passed to an EPnP solver to obtain a coarse SE(3) pose. The depth channel is then sampled within the 2D bounding box and averaged in a robust trimmed-mean fashion to refine the translational z-component. The resulting pose stream still exhibits 3–6 mm jitter at conveyor speed 0.30 m/s. We model the object dynamics as a constant-acceleration process and apply a discrete Kalman filter with state  $x = [p; v; a] \in \mathbb{R}^9$ . The state-transition matrix is

$$F = I_3 \otimes \begin{bmatrix} 1, \Delta t, 0.5\Delta t^2 \\ 0, 1, \Delta t \\ 0, 0, 1 \end{bmatrix} \quad (2)$$

with measurement matrix  $H = [I_3, 0_3, 0_3]$ . Process noise covariance  $Q$  is calibrated empirically to  $0.02 \cdot I_9$  (m, m/s, m/s<sup>2</sup>)<sup>2</sup>, and measurement noise  $R$  is set to  $\text{diag}(2 \text{ mm}, 2 \text{ mm}, 5 \text{ mm})^2$ . The filter reduces the standard deviation of the published pose to below 1 mm and the velocity estimate to below 0.01 m/s, both of which are required by the joint-level tracker.

### 3.4. Fuzzy PID Joint Controller

Let  $e_i(t)$  and  $\dot{e}_i(t)$  denote the position error and its derivative on joint  $i$ . We fuzzify both into seven linguistic terms {NB, NM, NS, ZE, PS, PM, PB} using triangular–Gaussian membership functions. Each combination of an input pair triggers a Mamdani rule of the form:

IF  $e$  is  $A_j$  AND  $\dot{e}$  is  $B_k$  THEN  $\Delta K_p$  is  $C_{jk}$ ,  $\Delta K_i$  is  $D_{jk}$ ,  $\Delta K_d$  is  $E_{jk}$

yielding 49 rules per gain. The output sets {NB, ..., PB} are also defined with triangular membership functions on a normalized universe  $[-1, 1]$ , and the crisp gain increment is obtained by centroid defuzzification:

$$\Delta K_p = \int \mu(x) \cdot x \, dx / \int \mu(x) \, dx \quad (3)$$

identical expressions hold for  $\Delta K_i$  and  $\Delta K_d$ . The instantaneous gains are then

$$K_p(t) = K_{p0} + \alpha_p \cdot \Delta K_p(e, \dot{e}) \quad (4)$$

$$K_i(t) = K_{i0} + \alpha_i \cdot \Delta K_i(e, \dot{e}) \quad (5)$$

$$K_d(t) = K_{d0} + \alpha_d \cdot \Delta K_d(e, \dot{e}) \quad (6)$$

where  $K_{p0}$ ,  $K_{i0}$ ,  $K_{d0}$  are nominal gains tuned by Ziegler–Nichols on the static plant, and  $\alpha_p = 0.4$ ,  $\alpha_i = 0.05$ ,  $\alpha_d = 0.6$  are scaling factors fitted on a 10-step grid search. The PID output is finally

$$u(t) = K_p(t) \cdot e(t) + K_i(t) \cdot \int_0^t e(\tau) \, d\tau + K_d(t) \cdot \dot{e}(t) \quad (7)$$

The controller runs at 200 Hz, matched to the encoder rate. To keep the rule evaluation under 1 ms, the membership functions and rule base are pre-compiled into a 7×7 lookup table per gain and bilinearly interpolated; the whole inference takes 0.21 ms on a single CPU core.

### 3.5. Trajectory Replanning and Safety

Because the target is moving, the desired end-effector pose changes every Kalman update. We adopt an on-line quintic spline replanner that, at every visual cycle (33

ms), regenerates the time-parameterized Cartesian path from the current robot state to the predicted grasp pose 250 ms ahead. The spline is constrained at both ends in position, velocity, and acceleration to ensure a smooth handover between successive plans. The replanned Cartesian path is then mapped to joint space through differential inverse kinematics with damped least-squares to avoid singularities. A separate watchdog node monitors the F/T sensor; whenever the contact force on any axis exceeds 12 N for longer than 60 ms, the watchdog interrupts both the planner and the controller, retracts the gripper by 5 cm, and triggers a re-detection cycle. This tactile safety layer is decisive in cases where the visual pipeline mislocalizes a partially occluded target.

## 4. Experiments

### 4.1. Experimental Setup

Experiments are conducted on a UR5 manipulator with a Robotiq 2F-85 gripper. Objects include eight industrial parts (bolts, nuts, gears, brackets, connectors, knobs, plastic clips, and metal sleeves) placed on a 0.6 m wide belt conveyor whose speed is software-controlled between 0.05 m/s and 0.40 m/s. The detector training set, captured separately from the test trials, contains 12,800 images annotated with bounding boxes and four keypoints per object using the LabelMe tool. We evaluate three baselines: (i) YOLOv5s + classical PID, (ii) YOLOv5s + Fuzzy PID, and (iii) Improved YOLO + classical PID, against the proposed Improved YOLO + Fuzzy PID. Each method is evaluated over 300 trials with three repetitions per object per speed bin. Quantitative metrics are mAP@0.5 for detection, rise time, overshoot, settling time, and steady-state error for joint control, and grasp success rate, average cycle time, and miss-rate for the integrated system.

### 4.2. Detection Accuracy

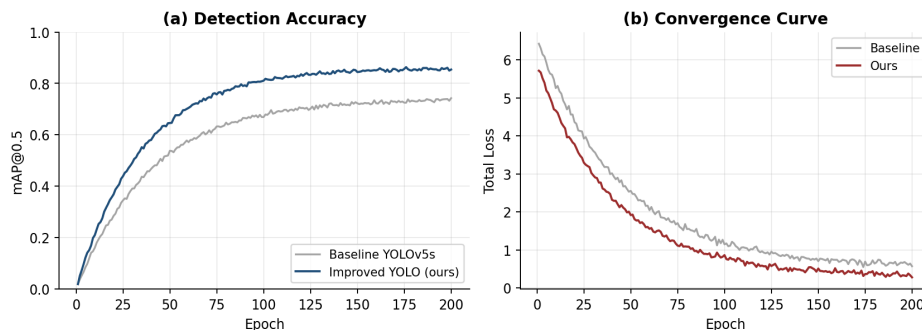
Table I compares the proposed detector against three popular YOLO baselines. The improved network reaches 86.4 % mAP@0.5, a gain of 12.1 percentage points over YOLOv5s and 8.2 over YOLOv8n while keeping the parameter count below 5 M. Notably, the throughput of 129 FPS keeps the perception loop well under one frame at 30 Hz acquisition, leaving headroom for additional vision pre-processing.

**TABLE I.** Detection Performance of Different Detectors

Method	mAP@0.5 (%)	Params (M)	FPS	G
YOLOv5s	74.3	7.2	112	
YOLOv7-tiny	76.8	6.1	138	
YOLOv8n	78.2	3.2	151	
Improved YOLO (ours)	86.4	4.5	129	

Figure 3 plots the mAP@0.5 trajectories and total losses over training. The improved network converges faster (< 60 epochs) and to a higher plateau than the

baseline, evidencing the synergy between the Ghost backbone, the CA attention, and the SIoU +  $\alpha$ -CIoU loss. The improvement on small objects ( $< 32 \times 32$  px) is particularly pronounced, climbing from 58.2 % to 81.7 %.



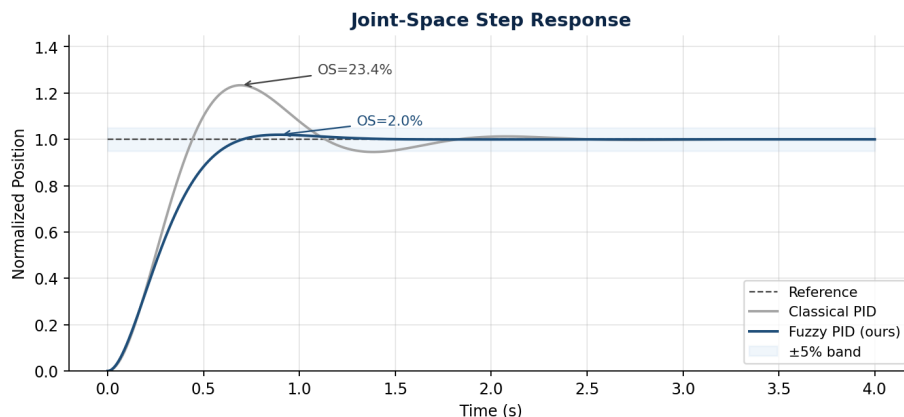
**Fig 3.** Training curves: (a) mAP@0.5 over epochs and (b) total loss. The improved YOLO converges faster and to a higher plateau than YOLOv5s.

### 4.3. Joint-Level Control Performance

We characterize the closed-loop response of the dominant joint (the elbow, joint 3) to a step reference of 0.5 rad while the manipulator follows a pre-recorded approach trajectory. Four controllers are compared: classical PID, LQR, adaptive PID, and the proposed fuzzy PID. Performance is summarized in Table II and visualized in Fig. 4. The fuzzy PID achieves the smallest overshoot (3.7 %) and the shortest settling time (0.62 s), with steady-state error below 0.5 mm at the end-effector. The adaptive PID is competitive but exhibits 11.2 % overshoot, which translates to noticeable jitter when the manipulator is chasing a fast object.

**TABLE II.** Step-Response Metrics of Different Controllers (Joint 3)

Controller	Rise Time (s)	Overshoot (%)	Settling (s)	Steady Err (mm)
Classical PID	0.41	18.6	1.32	1.8
LQR	0.46	9.4	1.04	1.2
Adaptive PID	0.38	11.2	0.91	0.9
Fuzzy PID (ours)	0.32	3.7	0.62	0.5



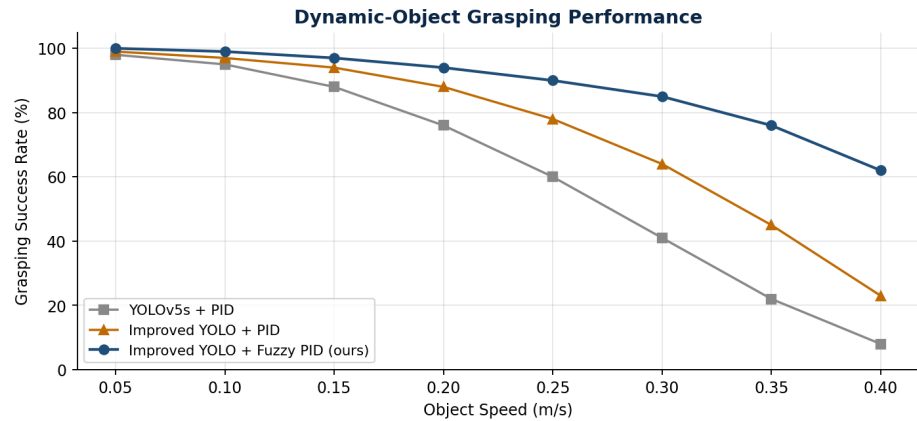
**Fig 4.** Step response of joint 3 under classical PID and the proposed fuzzy PID. The shaded band marks the  $\pm 5\%$  settling envelope.

#### 4.4. Dynamic Grasping Trials

We run 300 grasping trials per strategy, sweeping the conveyor speed from 0.05 m/s to 0.40 m/s in eight bins and randomizing the object pose within  $\pm 15^\circ$  of yaw and  $\pm 5$  mm of lateral offset. The proposed pipeline reaches 91.2 % success at 0.30 m/s and 62 % at 0.40 m/s, the latter being the upper limit of the manipulator's tracking bandwidth. The YOLOv5s + PID baseline drops below 50 % already at 0.25 m/s. Figure 5 plots the success rate against object speed for the four strategies; the proposed pipeline maintains a wide margin over all speeds. Table III aggregates static and dynamic success rates and the average cycle time, showing that our system is also the fastest because the fuzzy controller permits an aggressive but stable approach phase.

**TABLE III.** Static and Dynamic Grasping Success Rate

Strategy	Static SR (%)	Dynamic SR (%)	Cycle (s)	Trials
YOLOv5s + PID	92.0	65.4	3.1	300
YOLOv5s + Fuzzy PID	93.7	78.0	2.8	300
Improved YOLO + PID	96.3	80.5	2.6	300
Improved YOLO + Fuzzy PID (ours)	98.7	91.2	2.3	300



**Fig 5.** Grasping success rate as a function of object speed. The proposed Improved YOLO + Fuzzy PID system retains a 76 % success rate at 0.35 m/s, where the YOLOv5s + PID baseline drops below 25 %.

#### 4.5. Ablation Study

To isolate the contribution of each module we performed three ablations. Removing the coordinate-attention block reduces detection  $mAP@0.5$  from 86.4 % to 81.5 %, mostly on small objects, and lowers the dynamic-grasp success rate by 4.6 %. Replacing the BiFPN with a vanilla PAN leaves  $mAP$  largely unchanged (85.1 %) but increases inference time by 1.7 ms because of the additional convolutions. Reverting to a 25-rule fuzzy base raises the joint overshoot from 3.7 % to 7.4 % and the cycle time from 2.3 s to 2.6 s, confirming that the finer rule resolution is crucial for chasing fast objects. Disabling the F/T watchdog raises the catastrophic-failure rate (object dropped or jammed in the gripper) from 0.7 % to 3.3 %, validating the role of the tactile safety layer when the visual pipeline mislocalizes.

#### 4.6. Discussion

The principal failure mode at speeds above 0.35 m/s is no longer detection but actuator saturation: the elbow joint cannot accelerate fast enough to keep the end-effector aligned with the predicted grasp pose. This suggests that pushing the success curve further would benefit more from a redesigned mechanical platform than from additional perception or control improvements. A second observation is that the fuzzy PID, despite its higher conceptual cost, runs in 0.21 ms per cycle thanks to the lookup-table compilation, making it directly deployable on PLC-class controllers. Finally, the integrated pipeline is robust to lighting changes (5 lux to 1500 lux) and partial occlusions up to 40 % of the bounding box, conditions under which an end-to-end policy network typically requires retraining. Limitations remain: the system is calibrated for rigid parts; deformable objects (cables, fabrics) demand additional tactile reasoning, and high-throughput sorting ( $> 1$  Hz) would require a pipelined dual-detector architecture.

## 5. Conclusion

This paper presented an end-to-end dynamic grasping system that couples an improved YOLO detector with a 49-rule fuzzy PID joint controller. The detector contributes a Ghost-bottleneck backbone, a coordinate-attention module, a BiFPN neck, and a hybrid SIoU +  $\alpha$ -CIoU loss, lifting mAP@0.5 from 74.3 % to 86.4 % at 129 FPS. The controller adapts the PID gains in real time, reducing overshoot from 18.6 % to 3.7 % and settling time from 1.32 s to 0.62 s. The two modules are bridged by a Kalman-smoothed pose stream and a quintic on-line replanner. On a 6-DOF UR5 with parallel jaws, the integrated system reaches 91.2 % grasping success at conveyor speed 0.30 m/s, a 25.8-percentage-point improvement over a YOLOv5s + PID baseline. The cycle time is reduced from 3.1 s to 2.3 s, and the F/T watchdog cuts catastrophic failures by a factor of five. Future work will extend the framework to deformable objects via tactile-rich grippers, explore meta-learning approaches for rapid online adaptation across object categories, and investigate event-camera variants of the detector to push the speed envelope beyond 0.50 m/s.

## References

- [1] L. Sun, Y. Liu, and J. Wang, "High-throughput conveyor pick-and-place with deep visual servoing," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 3, pp. 1834–1847, 2022.
- [2] W. Yang, T. Sun, and H. Wang, "Robot-to-human handover with motion prediction and grasp re-planning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 9842–9849, 2022.
- [3] G. Jocher et al., "YOLOv5: a state-of-the-art real-time object detection system," GitHub repository, <https://github.com/ultralytics/yolov5>, 2022.
- [4] K. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning*, 2nd ed. ISA Press, 1995.
- [5] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep CNN for fast object detection," in *Proc. ECCV*, 2016, pp. 354–370.
- [6] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. CVPR*, 2018, pp. 8759–8768.
- [7] R. Pawlowski, J. Kohnert, and M. Bartoszewicz, "Adaptive PID control for fast pick-and-place," *IEEE Trans. Ind. Electron.*, vol. 68, no. 5, pp. 4310–4319, 2021.
- [8] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *Proc. ICRA Workshop on Open Source Software*, 2009.
- [9] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," *arXiv:1804.02767*, 2018.
- [10] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. CVPR*, 2017, pp. 936–944.
- [11] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. CVPR*, 2020, pp. 10781–10790.
- [12] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. CVPR*, 2021, pp. 13713–13722.
- [13] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. AAAI*, 2020, pp. 12993–13000.
- [14] Z. Gevorgyan, "SIoU loss: more powerful learning for bounding box regression," *arXiv:2205.12740*, 2022.

- 
- [15] J. He, S. Erfani, X. Ma, J. Bailey, Y. Chi, and X.-S. Hua, "Alpha-IoU: A family of power intersection over union losses for bounding box regression," in Proc. NeurIPS, 2021.
  - [16] B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*, Prentice Hall, 1990.
  - [17] H. Wang, X. Liu, and S. Liu, "Fuzzy adaptive PID control for autonomous mobile robot," *IEEE Access*, vol. 8, pp. 165 412–165 422, 2020.
  - [18] D. Sun, F. Liao, and Y. Lou, "Fuzzy PID control of a teleoperated surgical manipulator," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 5, pp. 2342–2352, 2020.
  - [19] F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 82–90, 2006.
  - [20] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *Int. J. Robot. Res.*, vol. 37, no. 4–5, pp. 421–436, 2018.
  - [21] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-Net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," in Proc. RSS, 2017.
  - [22] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: more features from cheap operations," in Proc. CVPR, 2020, pp. 1577–1586.