CoCoNav: Conformal Control for Safe Robot Navigation in Crowds

Cheng Guo[1,2], Mingzhe Ni[1] Zheng Liang[3], Yihu Ling[3], Yuan Hu[3], Michele Caprio[4], Daniele Pucci[1,5], Wei Pan[6]

***Abstract*— Safe and efficient robot navigation in crowds requires anticipating pedestrian motion despite uncertain and potentially shifting prediction errors. Existing reactive methods can produce oscillatory behavior, while predictive planners often treat forecasts as exact or rely on restrictive error models. Incorporating conservative uncertainty sets as hard constraints can also render model predictive control (MPC) infeasible. We propose *CoCoNav*, a crowd-navigation framework that combines online conformal calibration with runtime-certified planning. A horizon-specific conformal proportional–integral controller adapts trajectory-error bounds to regulate long-run empirical coverage, enabling the framework to respond to changing prediction errors. A *relax-then-verify* planner preserves solver feasibility by generating nominal trajectories with soft-constrained MPC and separately certifying them, together with contingency maneuvers, against the calibrated bounds before execution. Simulations and quadruped experiments show that CoCoNav achieves a favorable balance among collision avoidance, task success, and navigation efficiency relative to the evaluated baselines.**

I. INTRODUCTION

Safe and efficient robot navigation in crowds requires more than avoiding people at their currently observed positions: a robot must anticipate how pedestrians may move and account for the uncertainty of those predictions. This is difficult because human motion is interactive, multimodal, and nonstationary, so prediction errors can change as the crowd configuration and operating environment evolve [1]. The central challenge is therefore to convert fallible trajectory forecasts into actions that make progress without silently losing safety when the forecasts are inaccurate or the planning problem becomes difficult.

Reactive navigation methods compute collision-avoidance actions from the current configuration and remain attractive for their simplicity. Because they do not explicitly reason over future interactions, however, they may react late or behave conservatively in dense flows. Prediction-aware planners instead incorporate forecast pedestrian trajectories into trajectory optimization or model predictive control (MPC), allowing the robot to anticipate crossings and negotiate space over a planning horizon [2], [3]. Their effectiveness then depends not only on forecast accuracy, but also on how forecast errors are represented and used by the planner.

Prior work propagates prediction uncertainty through confidence-aware collision avoidance or stochastic MPC, typically using a specified probabilistic model of future motion [4], [5]. These methods are effective when that model remains representative, but model misspecification and distribution shift can invalidate the resulting uncertainty regions. Conformal prediction instead calibrates prediction regions from observed errors without prescribing a parametric error distribution [6]. Adaptive conformal control methods further update these regions online as errors arrive [7], [8]. Nevertheless, calibration only determines how much uncertainty to associate with a forecast; it does not determine how the robot should act when a calibrated region is too restrictive for the planner.

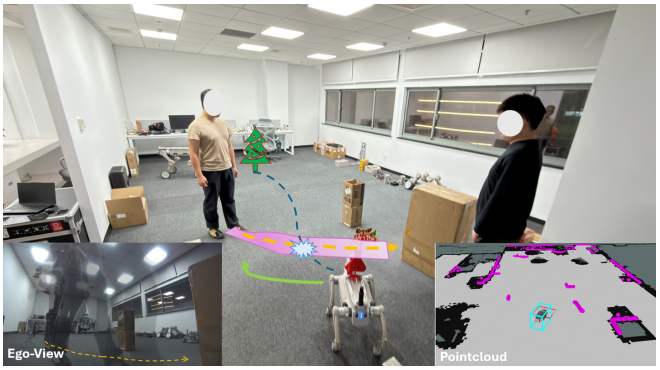


Fig. 1: Safe navigation around dynamic agents. (**Bottom Left**) Robot RGB ego-view showing the agent's directional motion trail. (**Bottom Right**) LiDAR point cloud. (**Main View**) The robot tracks the global reference (dotted blue) but deviates to a safe path (solid green) to avoid the agent's predicted trajectory (yellow) and conformal uncertainty sets (purple).

This planning challenge motivates a safety-filtering perspective that separates performance-oriented planning from runtime certification. A nominal controller first proposes an action or rollout, which is executed only if a monitoring condition certifies its safety; otherwise, the filter selects a contingency action [9], [10]. Existing predictive safety filters typically derive guarantees from known uncertainty models, robust reachable sets, or invariant terminal conditions. These assumptions are difficult to satisfy in crowd navigation because pedestrian prediction errors can evolve online with the surrounding scene. A practical filter must therefore distinguish a genuinely certified contingency from a fallback that is merely presumed safe, while also explicitly identifying cases in which no available candidate can be certified.

We address these coupled calibration, feasibility, and certification challenges with *CoCoNav*. Figure 1 illustrates how its calibrated prediction bounds inform crowd-avoidance behav-

[1]Department of Computer Science, The University of Manchester, Manchester, UK.
[2]Human-Robot Interfaces and Interaction Laboratory, Italian Institute of Technology, Genoa, Italy.
[3]Genisom AI, Shanghai, China.
[4]Department of Computer Science, University of Warwick, Coventry, UK.
[5]Generative Bionics, Genoa, Italy.
[6]School of Engineering, Newcastle University, Newcastle, UK.

ior. First, building on the proportional–integral mechanism of [8], we develop an online calibration scheme for pedestrian trajectory prediction. Each prediction horizon maintains an independent calibration state, and delayed prediction errors are evaluated against the thresholds associated with the forecasts that produced them. Second, a soft-constrained MPC incorporates the calibrated bounds as penalties to generate a progress-oriented nominal rollout, without interpreting the resulting solution as certified safe. Third, an *a posteriori* monitor certifies the entire nominal rollout against the same bounds. If certification fails, the monitor evaluates a library of dynamically admissible contingency trajectories and, if necessary, an acceleration-limited braking rollout before executing the first control input and replanning. By separating candidate generation from safety certification, CoCoNav avoids embedding conformal sets directly as hard MPC constraints. Moreover, rather than assuming the existence of an always-safe fallback, it explicitly records cases in which no available action can be certified. We integrate these components into an onboard pipeline and evaluate the resulting system in simulation and on a quadruped robot, providing initial evidence of its practical viability beyond simulation. Our main contributions are listed as follows:

- We adapt conformal proportional–integral (CPI) control to enable causal, horizon-specific calibration of multi-step pedestrian trajectory errors. The resulting uncertainty bounds adapt to evolving prediction errors while regulating long-run empirical coverage.
- We introduce a *relax-then-verify* control scheme in which a soft-constrained MPC generates candidate rollouts and an *a posteriori* module certifies nominal and contingency rollouts, mitigating hard-constraint infeasibility while retaining explicit statistical safety accounting.
- We evaluate the integrated system in extensive simulations and quadruped-robot experiments, providing evidence of its practical deployability and its safety–efficiency trade-off.

II. RELATED WORK

A. Prediction-Aware Crowd Navigation

Robot navigation among pedestrians has been studied through reactive, optimization-based, and learning-based approaches [11]. Classical reactive methods, including social-force models, dynamic-window search, and reciprocal velocity obstacles, construct collision-avoidance actions directly from the current configuration [12]–[14]. These methods are computationally attractive, whereas anticipatory navigation additionally requires reasoning about how pedestrian motion may evolve over the planning horizon.

Prediction-aware planners incorporate pedestrian forecasts into trajectory optimization or MPC [2], [3], [15], [16]. Learning-based alternatives obtain navigation policies from demonstrations [17]–[20] or reinforcement learning [21], [22]. In parallel, learned trajectory predictors model socially coupled and potentially multimodal pedestrian motion [23]–[25]. Regardless of the forecasting architecture, prediction errors remain consequential when a planner uses forecast positions to enforce robot–pedestrian clearance. Our work therefore focuses on online calibration of trajectory-prediction error rather than treating the output of a predictor as exact.

B. Online Conformal Calibratin for Motion Planning

Uncertainty-aware planners have represented prediction uncertainty using confidence regions, Gaussian models, and stochastic or chance-constrained optimization [4], [5], [26]–[28]. Their validity depends on the corresponding modeling and distributional assumptions. Conformal prediction instead constructs calibrated prediction regions from nonconformity scores without requiring a parametric error distribution [6]. Conformal regions have been incorporated into dynamic-environment planning and MPC constraints [29], [30], and have also been used to filter actions proposed by learned navigation policies [31]. Related decision-oriented formulations propagate conformal uncertainty into sensorimotor policy learning and downstream autonomous decisions [32], [33]. Standard split-conformal calibration is not designed to track a changing error sequence [34]. Adaptive conformal methods update their thresholds online [7], [35], [36], while Conformal PID control interprets coverage regulation through proportional, integral, and predictive feedback components [8]. We adapt its proportional–integral update to causal, horizon-indexed trajectory-error scores, resulting in a long-run empirical miscoverage bound under its stated conditions.

C. Runtime Certification for Safety Control

Calibrating predictive uncertainty does not by itself specify how a robot should act on the resulting uncertainty sets. One option is to impose them as hard planning constraints [29], [30], [35]; however, large online-calibrated sets can make a constrained finite-horizon problem infeasible. Safety filters offer a complementary performance–safety separation: a task controller proposes an action, while a runtime monitor accepts or modifies it using a safety-oriented fallback [9]. Within this taxonomy, model predictive shielding checks whether a candidate action followed by a fallback rollout can safely reach an invariant terminal set and switches to the fallback when the check fails [37]. Predictive safety filters instead compute a minimally modified input through finite-horizon optimization with terminal safety conditions [10]; probabilistic extensions replace worst-case tubes with probabilistic reachable sets [38]. Conformal predictive safety filtering provides a related statistical interface for learned controllers in dynamic environments [31]. These methods clarify the basis of safety claims, though their guarantees depend on the dynamics, uncertainty model, terminal safe set, and fallback availability. Our scheme evaluates nominal and contingency MPC rollouts using online-calibrated error bounds, braking if neither passes. Rather than enforcing an invariant terminal set, it supports a long-run clearance-violation bound that accounts for execution, perception, and no-certified-action events.

III. PROBLEM FORMULATION

We consider a robot governed by

$$\begin{aligned} x_{k+1} &= f(x_k, u_k), \qquad p_k = P(x_k) \in \mathbb{R}^2, \\ u_k &= (v_k, \omega_k) \in \mathcal{U}. \end{aligned} \tag{1}$$

where p_k indicates planar position and (v_k, ω_k) are the linear and angular velocities. We define a navigation task as $\mathcal{T} := (x_{\text{init}}, x_{\text{goal}}, \mathcal{O}, \mathcal{S}_o, \mathcal{S}_a, \mathcal{U}, \delta)$, where x_{init} and x_{goal} are the

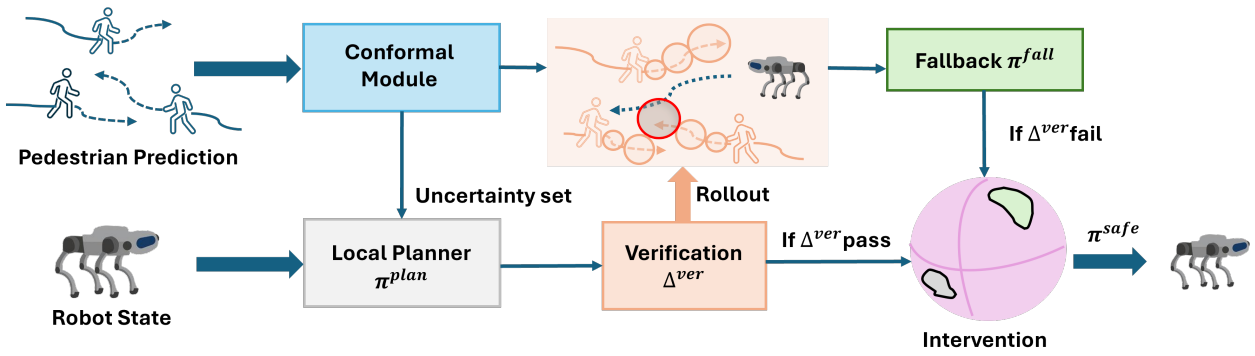


Fig. 2: Overview of CoCoNav. Online conformal calibration constructs adaptive robot no-go regions for soft MPC planning. A posteriori verification accepts the nominal roll-out or invokes the certified fallback policy, after which the first control of the selected roll-out is executed

initial and goal robot poses, $\mathcal{S}_o$ and $\mathcal{S}_a$ denote static obstacles and moving pedestrians, $\mathcal{U}$ is the robot action space, and $\delta \in (0,1)$ is the target conformal miscoverage rate. At each time step k, the observation $o_k = (o_k^{\mathrm{rgb}}, o_k^{\mathrm{geo}}) \in \mathcal{O}$ combines visual and geometric sensing.

Let $\mathcal{I}_k$ be the physical pedestrians relevant to safety, with positions $y_{k,j} \in \mathbb{R}^2$, and let $\mathcal{J}_k \subseteq \mathcal{I}_k$ be those currently tracked and available to the controller. We define the signed clearance from a robot position p as

$$c_j(p,y) := \|p-y\|_2 - (R_r + R_j), \tag{2}$$

where R_r and R_j are the robot and pedestrian radii. Given a required clearance $d_{\mathrm{safe}} \geq 0$, we have

$$h_k := \min_{j\in\mathcal{I}_k} c_j(p_k, y_{k,j}), \qquad V_k := \mathbf{1}\{h_k < d_{\mathrm{safe}}\}. \tag{3}$$

The objective is to reach x_{goal} while avoiding $\mathcal{S}_o \cup \mathcal{S}_a$ and controlling the long-run empirical frequency of V_k. In the ideal certified operating domain, the average nonviolation rate is at least $1-\delta$.

IV. PROPOSED METHOD

Figure 2 illustrates the closed-loop pipeline of CoCoNav. Tracked pedestrian histories produce point-trajectory forecasts, while delayed prediction errors update horizon-wise conformal thresholds that define adaptive robot no-go regions. A soft MPC planner generates a nominal roll-out, which is accepted only after posterior verification of dynamic conformal-clearance conditions. If verification fails, the same test is applied to the fallback action library and emergency-braking roll-out; the first control of the selected roll-out is then executed before replanning.

A. Pedestrian Trajectory Forecasting

For each pedestrian $j \in \mathcal{J}_k$, we denote its observed trajectory over L steps with

$$\mathbf{Y}_{k,j}^{\mathrm{hist}} := [y_{k-L+1,j}, \ldots, y_{k,j}]^\top \in \mathbb{R}^{L\times 2}. \tag{4}$$

We train a VAE-based predictor Φ_θ on the ETH/UCY datasets as in [25] and compute H-steps rollout in the future

$$\widehat{\mathbf{Y}}_{k,j} := \Phi_\theta(\mathbf{Y}_{k,j}^{\mathrm{hist}}) = [\widehat{y}_{k,j}^1, \ldots, \widehat{y}_{k,j}^H]^\top \in \mathbb{R}^{H\times 2}. \tag{5}$$

For compactness, let $Y_k := (y_{k,j})_{j\in\mathcal{J}_k}$ and $\widehat{Y}_k^\tau := (\widehat{y}_{k,j}^\tau)_{j\in\mathcal{J}_k}$ denote the joint observation and τ-step forecast under a fixed track ordering, and write $\widehat{Y}_k^{1:H} := (\widehat{Y}_k^1, \ldots, \widehat{Y}_k^H)$.

B. Trajectory-Error Conformal Calibration

1) Casual multi-step scores: For each horizon $\tau \in \{1,\ldots,H\}$, the τ-step-ahead forecast $\widehat{Y}_{k-\tau}^\tau$, issued at time $k-\tau$ for target time k, is evaluated once Y_k becomes available. The resulting time-lagged score updates the conformal state used to issue q_{k+1}^τ for the new forecast $\widehat{Y}_k^\tau$. Let $\mathcal{M}_k^\tau \subseteq \mathcal{J}_{k-\tau}$ contain the consistently matched tracks. For any fixed ordering of these tracks, we define

$$d_{k,j}^{\tau,\mathrm{pred}} := \|y_{k,j} - \widehat{y}_{k-\tau,j}^\tau\|_2, \tag{6}$$

$$\mathbf{z}_k^\tau := \mathrm{col}_{j\in\mathcal{M}_k^\tau}(y_{k,j} - \widehat{y}_{k-\tau,j}^\tau),$$

$$s_k^\tau := \|\mathbf{z}_k^\tau\|_2 = \left(\sum_{j\in\mathcal{M}_k^\tau} (d_{k,j}^{\tau,\mathrm{pred}})^2\right)^{1/2}. \tag{7}$$

This stacked Euclidean score is premutation invariant and directly satisfies the score-dominance relation

$$d_{k,j}^{\tau,\mathrm{pred}} \leq s_k^\tau, \qquad j \in \mathcal{M}_k^\tau. \tag{8}$$

Each forecast, track set, and associated threshold is stored when issued. Missing or unmatched safety-relevant pedestrians are charged separately to the perception exception term.

2) Conformal quantile update: For every τ, the threshold $q_{k+1}^{\tau,\mathrm{raw}}$ is issued at planning step k together with $\widehat{Y}_k^\tau$. Therefore, the delayed score s_k^τ, which evaluates the forecast issued at $k-\tau$, is compared with the stored threshold $q_{k-\tau+1}^{\tau,\mathrm{raw}}$. Once the score is observed, set

$$e_k^{\tau,\mathrm{raw}} := \mathbf{1}\{s_k^\tau > q_{k-\tau+1}^{\tau,\mathrm{raw}}\}, \qquad g_k^\tau := e_k^{\tau,\mathrm{raw}} - \delta,$$

$$G_k^\tau := \begin{cases} \sum_{i=\tau}^k g_i^\tau, & k \geq \tau, \\ 0, & k < \tau. \end{cases} \tag{9}$$

Each horizon maintains an independent conformal state

$$P_{k+1}^\tau = P_k^\tau + \eta_k^\tau g_k^\tau, \tag{10}$$

$$q_{k+1}^{\tau,\mathrm{raw}} = P_{k+1}^\tau + r_k(G_{k-1}^\tau), \tag{11}$$

$$r_k(x) := K_I \widetilde{\tan}\left(\frac{x\log(k+1)}{C_{\mathrm{sat}}(k+1)}\right), \tag{12}$$

where $\widetilde{\tan}(z)$ equals $\tan(z)$ on $(-\pi/2, \pi/2)$ and saturates to the corresponding signed infinity outside this interval. The optional proportional scaling sets η_k^τ to the base learning rate times the recent finite score range. Since the raw recurrence may be negative, verification uses the physical budget

$$q_k^{\tau,\mathrm{eff}} := \max\{0, q_k^{\tau,\mathrm{raw}}\}. \tag{13}$$

Consequently, $\mathbf{1}\{s_k^\tau > q_{k-\tau+1}^{\tau,\mathrm{eff}}\} \le e_k^{\tau,\mathrm{raw}}$. The region issued at time k for horizon τ is the joint norm ball

$$C_k^\tau := \left\{ (y_j)_{j\in\mathcal{J}_k} : \left(\sum_{j\in\mathcal{J}_k} \|y_j - \widehat{y}_{k,j}^\tau\|_2^2 \right)^{1/2} \le q_{k+1}^{\tau,\mathrm{eff}} \right\}. \tag{14}$$

To obtain the final clearance-violation guarantee, we can prove the following proposition that bounds the empirical one-step miscoverage frequency.

Proposition 1 (One-sided empirical calibration)**.** *Assume $K_I, C_{\mathrm{sat}} > 0$, all scores and learning rates are finite. For*

$$a_t := \frac{\pi C_{\mathrm{sat}}}{2} \frac{t+1}{\log(t+1)}, \qquad \bar{a}_t := \max_{1\le i\le t} a_i,$$

the one-step stream ($\tau = 1$) satisfies, for $T \ge 2$,

$$\frac{1}{T}\sum_{t=1}^{T} e_t^{1,\mathrm{raw}} \le \delta + r_T, \qquad r_T := \frac{\bar{a}_{T-1} + 3}{T} \longrightarrow 0. \tag{15}$$

Proof. For $\tau = 1$, saturation makes $q_{t+1}^{1,\mathrm{raw}} = +\infty$ whenever $G_{t-1}^1 \ge a_t$, so the next finite score is covered. The one-round delay permits an overshoot of at most two, and induction gives $G_t^1 \le \bar{a}_{t-1} + 2$. Accounting for initialization adds at most one; using $G_T^1 = \sum_{t=1}^T (e_t^{1,\mathrm{raw}} - \delta)$ yields (15). Finally, $\bar{a}_T = O(T/\log T)$ implies $r_T \to 0$. ∎

C. Uncertainty-Aware MPC

1) Nominal candidate: At time k, the local planner computes a nominal sequence $U_k = (u_{k|k}, \dots, u_{k+H-1|k})$ and state roll-out $X_k = (x_{k|k}, \dots, x_{k+H|k})$ by solving

$$\min_{X_k, U_k} \quad J_{\mathrm{track}}(X_k, U_k) + \sum_{\tau=1}^{H} \sum_{j\in\mathcal{J}_k} \psi\big(d_{k,j}^\tau\big) \tag{16a}$$

$$\text{s.t.} \quad x_{k|k} = x_k,$$
$$x_{k+\tau+1|k} = f(x_{k+\tau|k}, u_{k+\tau|k}), \quad \tau = 0, \dots, H-1,$$
$$u_{k+\tau|k} \in \mathcal{U}, \quad \tau = 0, \dots, H-1, \tag{16b}$$
$$d_{k,j}^\tau := c_j(p_{k+\tau|k}, \widehat{y}_{k,j}^\tau) - q_{k+1}^{\tau,\mathrm{eff}}. \tag{16c}$$

Here J_{track} collects path tracking and velocity control terms, ψ is a decreasing smooth penalty.

2) A posteriori certification: Let $Z \in \mathcal{Z}_k$ denote a dynamically feasible horizon-H robot roll-out from x_k, with predicted position $p_{k+\tau|k}^Z$ at horizon τ. Any nominal or fallback roll-out Z is certified only if

$$c_j(p_{k+\tau|k}^Z, \widehat{y}_{k,j}^\tau) \ge d_{\mathrm{safe}} + q_{k+1}^{\tau,\mathrm{eff}} + \varepsilon, \quad {}^{\tau=1,\dots,H,}_{j\in\mathcal{J}_k}, \tag{17}$$

Algorithm 1 Trajectory-Error CPI–MPC Pipeline

Require: $\delta, H, d_{\mathrm{safe}}, \varepsilon$ and CPI/fallback states
Observe (x_k, Y_k) and retrieve the forecast archive
Evaluate available delayed scores s_k^τ by (7)
Update $\{P_{k+1}^\tau, G_k^\tau, q_{k+1}^{\tau,\mathrm{eff}}\}_{\tau=1}^H$ by (9)–(13)
Predict $\widehat{Y}_k^{1:H}$ using Φ_θ
Solve the soft MPC (16) for (X_k^*, U_k^*)
$Z_k^{\mathrm{nom}} \leftarrow \mathrm{Rollout}(x_k, U_k^*)$
if $\Gamma_k(Z_k^{\mathrm{nom}}) = 1$ **then**
  $Z_k^{\mathrm{sel}} \leftarrow Z_k^{\mathrm{nom}}$
else
  Verify the library: $\mathcal{C}_k^{\mathrm{lib}} \leftarrow \{Z \in \mathcal{Z}_k^{\mathrm{lib}} : \Gamma_k(Z) = 1\}$
  if $\mathcal{C}_k^{\mathrm{lib}} \neq \varnothing$ **then**
    $Z_k^{\mathrm{sel}} \leftarrow \arg\max_{Z\in\mathcal{C}_k^{\mathrm{lib}}} [m_k(Z) - J_{\mathrm{sec}}(Z)]$
  else
    $Z_k^{\mathrm{sel}} \leftarrow Z_k^{\mathrm{br}}$
  end if
end if
$N_k \leftarrow \mathbf{1}\{\Gamma_k(Z_k^{\mathrm{sel}}) = 0\}$
Archive the issued forecasts, identities, and thresholds
Execute $u_k \leftarrow \Pi_0(Z_k^{\mathrm{sel}})$ and replan

where $\varepsilon \ge 0$ is a verification reserve. We then define the worst-case certified margin and validity indicator

$$\mu_{k,j}^\tau(Z) := c_j(p_{k+\tau|k}^Z, \widehat{y}_{k,j}^\tau) - q_{k+1}^{\tau,\mathrm{eff}} - d_{\mathrm{safe}} - \varepsilon, \tag{18}$$

$$m_k(Z) := \min_{\substack{1\le\tau\le H \\ j\in\mathcal{J}_k}} \mu_{k,j}^\tau(Z), \tag{19}$$

$$\Gamma_k(Z) := \mathbf{1}\{m_k(Z) \ge 0\}. \tag{20}$$

Let N_k indicate that neither the nominal, nor the fallback action can be certified, and B_{k+1} for any missing pedestrian tracks. Algorithm 1 summarizes the receding-horizon loop, including conformal updates for trajectory prediction, soft MPC and posterior certification. The full-horizon test is operationally conservative, and the theorem below certifies only the first control that is actually executed.

Theorem 1 (Fallback-aware clearance bound)**.** *Under Proposition 1, causal forecast logging, and* (8)*, Algorithm 1 satisfies*

$$\frac{1}{T}\sum_{k=0}^{T-1} V_{k+1} \le \delta + r_T + \frac{1}{T}\sum_{k=0}^{T-1} (B_{k+1} + N_k). \tag{21}$$

Proof. On a certified round with $s_{k+1}^1 \le q_{k+1}^{1,\mathrm{eff}}$ and $B_{k+1} = 0$, (8) gives $\|y_{k+1,j} - \widehat{y}_{k,j}^1\|_2 \le q_{k+1}^{1,\mathrm{eff}}$ for every safety-relevant pedestrian. The one-step case of (17) and the reverse triangle inequality give

$$c_j(p_{k+1}, y_{k+1,j}) \ge c_j(p_{k+1|k}^{\mathrm{sel}}, \widehat{y}_{k,j}^1) - q_{k+1}^{1,\mathrm{eff}} \ge d_{\mathrm{safe}} + \varepsilon.$$

Hence, we have pathwise $V_{k+1} \le e_{k+1}^{1,\mathrm{raw}} + B_{k+1} + N_k$. Summation and Proposition 1 prove (21). ∎

D. Certified Contingency Control

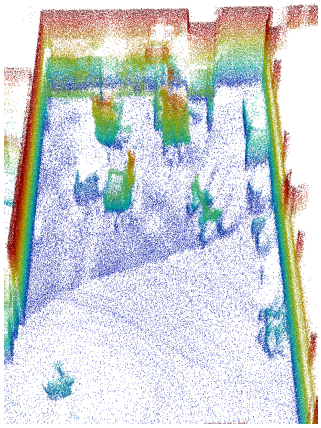
If the nominal roll-out fails certification, the controller verifies a finite library of hold, slow, and straight/turning



(a) The real-world environment.



(b) The simulated environment.



(c) The reconstructed environment with pointcloud.

Fig. 3: Snapshots of the environment at different levels.

brake roll-outs:

$$\mathcal{C}_k^{\text{lib}} = \{Z \in \mathcal{Z}_k^{\text{lib}} : \Gamma_k(Z) = 1\}. \tag{22}$$

The executed roll-out is selected by

$$Z_k^{\text{sel}} \in \begin{cases} \arg\max_{Z \in \mathcal{C}_k^{\text{lib}}} [m_k(Z) - J_{\text{sec}}(Z)], & \mathcal{C}_k^{\text{lib}} \neq \varnothing, \\ \{Z_k^{\text{br}}\}, & \mathcal{C}_k^{\text{lib}} = \varnothing, \end{cases} \tag{23}$$

where J_{sec} ranks already-certified candidates by speed and smoothness, and Z_k^{br} is an acceleration-limited stopping roll-out. Braking is certified only if $\Gamma_k(Z_k^{\text{br}}) = 1$; else $N_k = \mathbf{1}\{\Gamma_k(Z_k^{\text{sel}}) = 0\}$ records that no certified action exists.

V. EXPERIMENTS AND RESULTS

A. Simulation Environment

We evaluate the proposed framework in `MATRiX`[1], a high-fidelity simulation platform that integrates MuJoCo, Unreal Engine 5, and CARLA to provide interactive environments for robotics research. We design three crowd-navigation scenarios within a 10m×10m area, each populated with 4 to 12 pedestrians. In every scenario, the robot must traverse the area from bottom to top while avoiding both static obstacles and moving pedestrians. The three scenarios differ in the structure of the crowd flow: *Opposing Flow*, in which the robot navigates against a crowd moving in the opposite direction; *Intersecting Flow*, in which it crosses a stream of pedestrians moving perpendicular to its path; and *Unstructured Flow*, in which it navigates through a crowd with random, multi-directional motion. Figure 3 illustrates an example of the real and simulated environments.

B. Baselines

We select three baseline methods for comparison. Since our framework targets a real-world robotic pipeline, each baseline must be adapted to integrate with our perception and control stack. We therefore prioritize methods with open-source implementations that are compatible with our hierarchical navigation framework. The selected baselines are: **ST-Planner** [39], a spatiotemporal global planner that supports real-time replanning, with the resulting path tracked by a Pure Pursuit controller; **ACP-MPC** [35], an MPC-based local planner that incorporates adaptive conformal prediction; and **MPPI** [40], a sampling-based model predictive controller that computes commands from a cost-weighted average over multiple parallel trajectory rollouts.

[1]`https://github.com/zsibot/matrix`

C. Evaluation Metrics

We quantify the planners' performance in crowd navigation using the following metrics, as in [11]. **Success Rate** (SR) measures the percentage of episodes in which the robot reaches its goal within the specified time limit, i.e., $SR = N_{\text{success}}/N_{\text{total}}$. **Minimum Distance** (MD) measures the average minimum distance between the robot and pedestrians during the navigation task, i.e., $MD = (1/N_{\text{total}}) \sum_{k=1}^{N_{\text{total}}} D_k$. **Collision Rate** (CR) measures the percentage of episodes in which at least one collision (MD below the threshold) occurs, i.e., $CR = N_{\text{collision}}/N_{\text{total}}$. **Navigation Time** (NT) measures the average travel time to the goal across all test episodes, i.e., $NT = (1/N_{\text{total}}) \sum_{k=1}^{N_{\text{total}}} T_k$. **Path Length** (PL) measures the average traveled distance to the goal across all test episodes, i.e., $PL = (1/N_{\text{total}}) \sum_{k=1}^{N_{\text{total}}} L_k$. These metrics collectively reflect the effectiveness (SR), safety (MD, CR), and efficiency (NT, PL) of the planners.

D. Implementation Details

Our implementation of CoCoNav is written in C++/ROS2 and will be open sourced upon acceptance. Simulations are conducted on an Ubuntu 22.04 system equipped with an NVIDIA RTX 4090 GPU. For trajectory prediction, we employ a conditional variational autoencoder that models the future pedestrian motion. The network encodes each agent's motion history with a recurrent backbone and aggregates neighboring agents through an attention-based social pooling module, capturing interactions among pedestrians. Conditioned on these representations, the decoder samples plausible future trajectories from the learned latent space. We pair the predictor with downsampling and adopt an 8-step history window and a 12-step prediction horizon. Additionally, we port CPI to C++ and enable adaptive parameter tuning. We solve local planning via Ceres Pedestrian and robot radii are set to 0.25m and 0.35m, respectively.

E. Comparison Experiments

1) Quantitative analysis: Table I compares our method against the baselines across three scenarios and three crowd densities, with each method run for 10 trials per configuration and the start and goal positions sampled from specified regions. In sparse crowds (Ped=4), most planners attain a high SR with few collisions, yet our method already stands out, reaching a 100% SR at a 0% CR in all scenarios while keeping a large MD at a short NT and PL. As the density grows to 8 and 12 pedestrians, the baselines degrade sharply, whereas our method sustains an SR of at least 80% and a CR of at most 10% throughout; in Intersecting Flow at Ped=12, for instance, both ST-Planner and MPPI collapse to a mere 10% SR while our method still reaches 80%. Notably, the baselines fail for opposite reasons: ST-Planner relies on reactive replanning and thus incurs a high CR under dense

TABLE I: Quantitative comparison between our method and baselines across various simulated crowd scenarios.

Env	Method	Ped=4					Ped=8					Ped=12				
		SR ↑	MD(m) ↑	CR ↓	NT(s) ↓	PL(m) ↓	SR ↑	MD(m) ↑	CR ↓	NT(s) ↓	PL(m) ↓	SR ↑	MD(m) ↑	CR ↓	NT(s) ↓	PL(m) ↓
Opposing Flow	ST-Planner	100%	0.46±0.26	20%	34.45±3.80	14.95±0.34	100%	0.42±0.02	100%	37.87±0.94	15.04±0.57	100%	0.39±0.19	80%	32.75±1.36	14.32±0.33
	MPPI	100%	0.62±0.09	10%	32.03±4.78	14.89±0.68	80%	0.59±0.12	10%	36.82±5.46	15.01±0.72	80%	0.53±0.12	20%	39.77±5.67	16.08±1.29
	ACP-MPC	80%	1.02±0.21	0%	40.73±3.61	17.53±0.43	0%	0.61±0.02	0%	102.49±4.34	33.74±2.08	0%	0.62±0.04	0%	97.86±2.12	32.94±1.88
	Ours	100%	0.66±0.10	0%	36.75±2.28	16.57±0.69	90%	0.64±0.07	0%	39.43±2.87	15.36±0.42	100%	0.62±0.07	0%	31.00±4.26	14.83±0.83
Intersecting Flow	ST-Planner	100%	0.61±0.08	0%	33.74±1.75	14.74±0.38	30%	0.42±0.01	100%	43.76±1.98	15.83±0.50	10%	0.34±0.05	100%	45.15±3.68	15.63±0.23
	MPPI	80%	0.62±0.02	0%	37.99±2.73	15.70±0.67	10%	0.56±0.09	10%	48.08±3.61	16.23±0.34	10%	0.40±0.05	100%	43.73±1.70	15.30±0.25
	ACP-MPC	0%	0.66±0.16	10%	51.34±6.59	18.89±1.62	0%	0.54±0.09	10%	105.81±7.80	33.06±3.85	0%	0.50±0.12	20%	117.65±7.23	41.25±3.24
	Ours	100%	0.67±0.07	0%	31.32±0.87	14.78±0.11	80%	0.55±0.02	0%	42.75±10.12	15.62±1.23	80%	0.53±0.05	10%	42.44±3.21	15.36±0.30
Unstructured Flow	ST-Planner	100%	0.54±0.06	20%	32.04±2.55	14.27±1.07	100%	0.64±0.06	0%	33.47±3.03	14.74±0.41	30%	0.52±0.17	20%	40.43±1.81	15.17±0.36
	MPPI	80%	0.59±0.06	10%	35.38±3.54	15.28±0.62	20%	0.52±0.11	20%	48.74±4.66	15.94±0.66	10%	0.54±0.09	80%	42.12±2.46	15.28±0.29
	ACP-MPC	80%	0.82±0.28	0%	46.09±11.68	19.03±3.15	30%	0.49±0.31	20%	58.40±16.08	20.61±2.87	0%	0.84±0.14	0%	81.99±8.20	23.90±4.25
	Ours	100%	0.79±0.13	0%	27.85±0.69	14.45±0.04	100%	0.52±0.06	10%	34.09±1.87	14.86±0.16	80%	0.58±0.08	0%	37.12±5.74	15.76±0.67

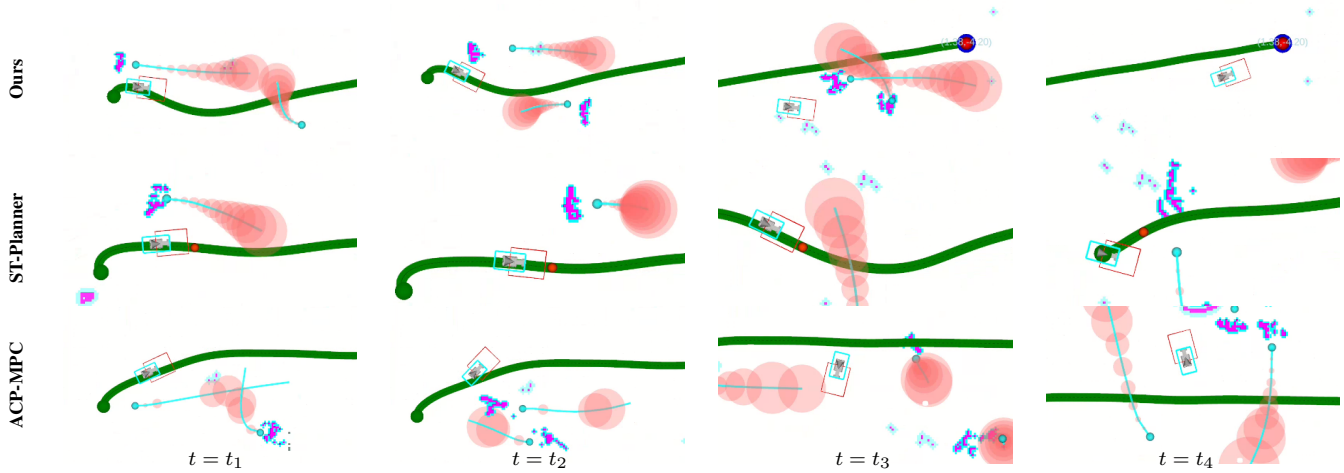


Fig. 4: Sequential snapshots of robot behaviors produced by our approach and baselines when navigating through crowds. The timestamps at the bottom indicate different physical time for each navigation episode.

flows (e.g., 100% CR in Opposing and Intersecting Flow at Ped=8), whereas ACP-MPC keeps CR low but grows overly conservative, inflating NT beyond twice ours (e.g., 102.49s vs. 39.43s in Opposing Flow at Ped=8) so that most episodes time out and its SR collapses to 0%. Our method avoids both failure modes, combining efficient navigation with a low collision rate.

2) Qualitative evaluation: Figure 4 presents sequential snapshots of the Unstructured Flow scenario (Ped=8), where the green curve traces the robot trajectory and the red regions denote the predicted pedestrian occupancy, while Fig. 5 details the corresponding motion profiles for our method and ACP-MPC. As shown in the first row of Fig. 4, our method anticipates the crowd and initiates early, smooth avoidance, staying close to the global plan and reaching the goal by t_4. This is corroborated by the top row of Fig. 5, where the executed trajectory closely follows the global plan and the predominantly forward linear velocity together with bounded angular commands ($|\omega| \leq 0.6$ rad/s) drive the robot to the goal in about 42 s. In contrast, ST-Planner reacts only to instantaneous observations, so pedestrians intrude directly onto its tracked path and induce a near-collision. ACP-MPC instead produces excessively large conformal sets that force overly conservative maneuvers; accordingly, the bottom row of Fig. 5 exhibits a pronounced detour loop and highly oscillatory velocity commands, requiring over 100 s to complete the episode.

F. Ablation Studies

To assess the individual contributions of our framework, we conduct two sets of ablation studies designed to answer the following questions: (1) Can CPI effectively handle distribution shifts? (2) Does the *relax-then-verify* strategy achieve an effective balance between safety and feasibility?

1) On CPI calibration: Figure 6 plots the per-frame quantification error ($r - \varepsilon$, the gap between the realized nonconformity score and the estimated quantile), and Table II summarizes its magnitude across horizons. Under the shift, ACP repeatedly diverges, saturating to the infinite-uncertainty bound (top ticks of the middle panel); we thus report ACP* over the finite frames only, which still understates its true error. The two bounded schemes are nearly identical at h=1, but CPI adapts better as the horizon grows, attaining a tighter error at h=12 (0.463 vs. 0.543), where its curve stays centered near zero while Conformal P-control drifts to larger margins. Table III shows the downstream effect on local planning (Unstructured Flow, Ped=8). ACP's divergence cripples its planner: the oversized sets repeatedly render the optimization infeasible, limiting ACP-MPC to a 20% SR at an

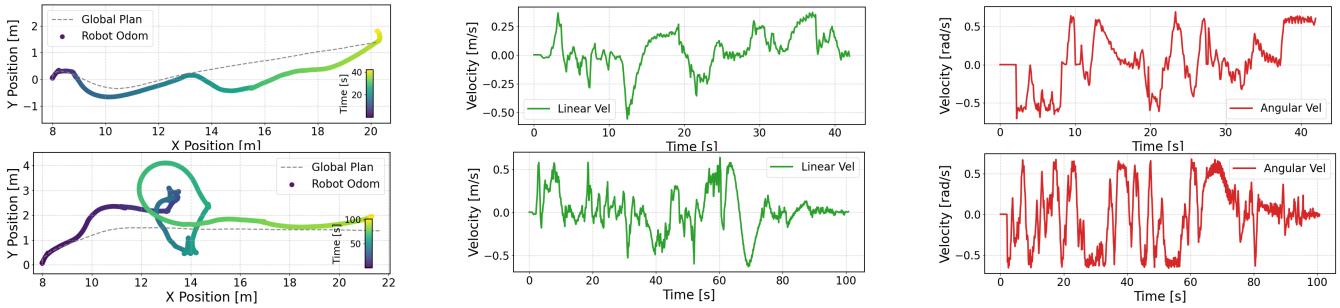


Fig. 5: Comparison of robot motion generation between our method (top) and ACP-MPC (bottom): trajectory (left), linear velocity (middle), and angular velocity (right).

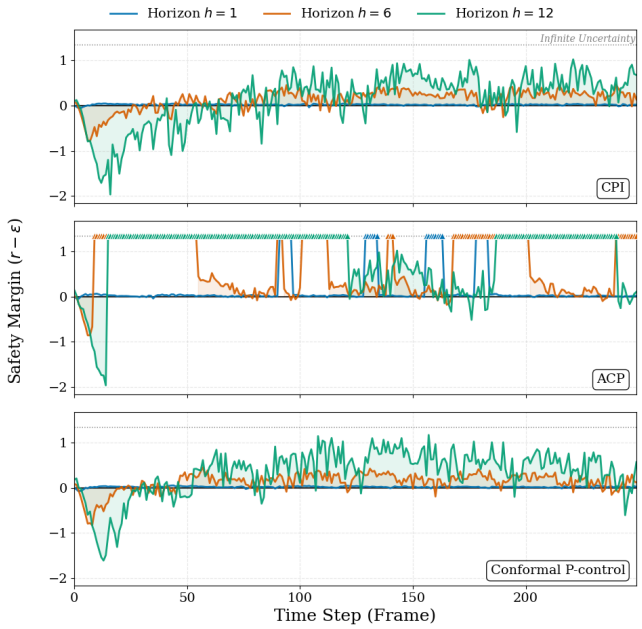


Fig. 6: Ablation of conformal safety margins under distribution shift. The margin is the actual nonconformity score minus the estimated quantile.

TABLE II: Statistical results of conformal safety marginals.

Method	Horizon		
	h=1	h=6	h=12
ACP*	0.022±0.014	0.183±0.124	0.386±0.277
Conformal P-control	0.022±0.010	0.205±0.108	0.543±0.272
CPI	0.023±0.010	0.225±0.104	0.463±0.249

inflated 58.64s navigation time. In contrast, both Conformal P-control MPC and CPI-MPC reach a 100% SR, confirming that bounded, well-calibrated uncertainty is essential for reliable planning under domain shift, with CPI-MPC being slightly more efficient (28.83s, 13.90m).

2) On relax-then-verify control: Table IV isolates our two-stage control mechanism in a medium-density scenario (Ped=8), comparing against Quasi-Hard MPC, which enforces the constraints via high cost penalties, and Soft-Only MPC, which relaxes them but omits the posterior verification.

TABLE III: Crowd navigation performance with different conformal prediction methods (unstructured flow, Ped=8).

Method	SR ↑	MD(m) ↑	CR ↓	NT(s) ↓	PL(m) ↓
ACP-MPC	20%	0.60±0.18	10%	58.64±7.83	21.18±2.35
Conformal P-control MPC	100%	0.58±0.08	20%	29.50±1.80	15.20±0.90
CPI-MPC	100%	0.56±0.04	20%	28.83±1.48	13.90±0.42

TABLE IV: Ablation results of *relax-then-verify* strategy.

Method	**Components**		**Metrics**		
	Relaxed?	Verify?	MD(m) ↑	NT(s) ↓	PL(m) ↓
Quasi-Hard MPC	✗	✗	0.65±0.28	46.82±13.91	19.03±3.44
Soft-Only MPC	✓	✗	0.52±0.08	26.75±1.59	13.35±0.38
CPI-MPC	✓	✓	0.56±0.04	28.83±1.48	13.90±0.42

Enforcing hard constraints leaves Quasi-Hard MPC overconservative and erratic, nearly doubling the navigation time (46.82s) with a large variance (±13.91). Relaxation alone (Soft-Only MPC) restores efficiency (26.75s) but yields the lowest safety clearance (0.52m). Reinstating the verification step, our full scheme recovers the clearance to 0.56m at a marginal efficiency cost (28.83s), while attaining the smallest variance on every metric. These results confirm that the *relax-then-verify* strategy effectively balances safety and feasibility.

G. Real-World Deployment

We deploy our approach on a 12-DoF quadruped robot equipped with a Mid-360 LiDAR and an IMX415 camera module. The perception module fuses these two sensing modalities: the LiDAR provides odometry and a real-time occupancy map of static obstacles, while the camera stream is processed by YOLOv11 [41] for pedestrian detection and ByteTrackV2 [42] for multi-object tracking. The resulting detections are back-projected with the LiDAR depth and transformed into the global frame to recover each pedestrian's position and track history, which are then fed to the trajectory predictor. This perception module runs on an RK3588-based processing board (4× Cortex-A76 cores at 2.4GHz paired with 4× Cortex-A55 cores at 1.8GHz) at 10Hz, whereas the prediction and navigation modules run on an NVIDIA Xavier NX board, with the planner operating at 50Hz. More details are presented in the supplementary video.

VI. CONCLUSIONS

CoCoNav acts on fallible pedestrian forecasts by keeping online conformal calibration and runtime certification as separate stages, rather than embedding uncertainty sets as hard constraints. This decoupling preserves progress as crowd density grows, sustaining a high success rate with near-zero collisions while avoiding the detours and stalls that make conservative conformal-constrained MPC infeasible. Ablations confirm that horizon-specific PI calibration yields tighter margins under distribution shift and that a posteriori verification recovers the safety clearance lost by soft-only planning, with onboard quadruped deployment indicating these behaviors transfer to a real robot. A current limitation is that calibration trusts the prediction stage alone, ignoring perception and state-estimation noise. A promising direction is to fold latent world models into the loop, producing and acting on conformally bounded predictions within a single, tightly integrated decision process.

REFERENCES

[1] A. Francis, C. Pérez-d'Arpino, C. Li, F. Xia, A. Alahi, R. Alami, A. Bera, A. Biswas, J. Biswas, R. Chandra *et al.*, "Principles and guidelines for evaluating social robot navigation algorithms," *ACM Transactions on Human-Robot Interaction*, vol. 14, no. 2, pp. 1–65, 2025.

[2] B. Brito, B. Floor, L. Ferranti, and J. Alonso-Mora, "Model predictive contouring control for collision avoidance in unstructured dynamic environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4459–4466, 2019.

[3] S. Poddar, C. Mavrogiannis, and S. S. Srinivasa, "From crowd motion prediction to robot navigation in crowds," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 6765–6772.

[4] D. Fridovich-Keil, A. Bajcsy, J. F. Fisac, S. L. Herbert, S. Wang, A. D. Dragan, and C. J. Tomlin, "Confidence-aware motion prediction for real-time collision avoidance1," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 250–265, 2020.

[5] S. H. Nair, V. Govindarajan, T. Lin, C. Meissen, H. E. Tseng, and F. Borrelli, "Stochastic mpc with multi-modal predictions for traffic intersections," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 635–640.

[6] A. N. Angelopoulos, S. Bates *et al.*, "Conformal prediction: A gentle introduction," *Foundations and trends® in machine learning*, vol. 16, no. 4, pp. 494–591, 2023.

[7] I. Gibbs and E. Candes, "Adaptive conformal inference under distribution shift," *Advances in Neural Information Processing Systems*, vol. 34, pp. 1660–1672, 2021.

[8] A. Angelopoulos, E. Candes, and R. J. Tibshirani, "Conformal pid control for time series prediction," *Advances in neural information processing systems*, vol. 36, pp. 23 047–23 074, 2023.

[9] K.-C. Hsu, H. Hu, and J. F. Fisac, "The safety filter: A unified view of safety-critical control in autonomous systems," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 7, 2023.

[10] K. P. Wabersich and M. N. Zeilinger, "A predictive safety filter for learning-based control of constrained nonlinear dynamical systems," *Automatica*, vol. 129, p. 109597, 2021.

[11] R. Mirsky, X. Xiao, J. Hart, and P. Stone, "Conflict avoidance in social navigation—a survey," *ACM Transactions on Human-Robot Interaction*, vol. 13, no. 1, pp. 1–36, 2024.

[12] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.

[13] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE robotics & automation magazine*, vol. 4, no. 1, pp. 23–33, 2002.

[14] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics Research: The 14th International Symposium ISRR*. Springer, 2011, pp. 3–19.

[15] O. De Groot, L. Ferranti, D. M. Gavrila, and J. Alonso-Mora, "Topology-driven parallel trajectory optimization in dynamic environments," *IEEE Transactions on Robotics*, 2024.

[16] D. Martinez-Baselga, O. de Groot, L. Knoedler, L. Riazuelo, J. Alonso-Mora, and L. Montano, "Shine: Social homology identification for navigation in crowded environments," *The International Journal of Robotics Research*, p. 02783649251344639, 2024.

[17] Z. Xie, P. Xin, and P. Dames, "Towards safe navigation through crowded dynamic environments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 4934–4940.

[18] A. Pokle, R. Martín-Martín, P. Goebel, V. Chow, H. M. Ewald, J. Yang, Z. Wang, A. Sadeghian, D. Sadigh, S. Savarese *et al.*, "Deep local trajectory replanning and control for robot navigation," in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 5815–5822.

[19] H. Karnan, A. Nair, X. Xiao, G. Warnell, S. Pirk, A. Toshev, J. Hart, J. Biswas, and P. Stone, "Socially compliant navigation dataset (scand): A large-scale dataset of demonstrations for social navigation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 807–11 814, 2022.

[20] R. Chandra, H. Karnan, N. Mehr, P. Stone, and J. Biswas, "Towards imitation learning in real world unstructured social mini-games in pedestrian crowds," *arXiv e-prints*, pp. arXiv–2405, 2024.

[21] C. Chen, Y. Liu, S. Kreiss, and A. Alahi, "Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning," in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 6015–6022.

[22] M. Everett, Y. F. Chen, and J. P. How, "Collision avoidance in pedestrian-rich environments with deep reinforcement learning," *Ieee Access*, vol. 9, pp. 10 357–10 377, 2021.

[23] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 961–971.

[24] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *European Conference on Computer Vision*. Springer, 2020, pp. 683–700.

[25] P. Xu, J.-B. Hayet, and I. Karamouzas, "Socialvae: Human trajectory prediction using timewise latents," in *European Conference on Computer Vision*. Springer, 2022, pp. 511–528.

[26] D. D. Fan, K. Otsu, Y. Kubo, A. Dixit, J. Burdick, and A.-A. Agha-Mohammadi, "Step: Stochastic traversability evaluation and planning for risk-aware off-road navigation," *arXiv preprint arXiv:2103.02828*, 2021.

[27] M. Omainska, J. Yamauchi, T. Beckers, T. Hatanaka, S. Hirche, and M. Fujita, "Gaussian process-based visual pursuit control with unknown target motion learning in three dimensions," *SICE Journal of Control, Measurement, and System Integration*, vol. 14, no. 1, pp. 116–127, 2021.

[28] K. Nakamura and S. Bansal, "Online update of safety assurances using confidence-based predictions," *arXiv preprint arXiv:2210.01199*, 2022.

[29] L. Lindemann, M. Cleaveland, G. Shim, and G. J. Pappas, "Safe planning in dynamic environments using conformal prediction," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 5116–5123, 2023.

[30] Z. Huang, T. Ji, H. Zhang, F. C. Pouria, K. Driggs-Campbell, and R. Dong, "Interaction-aware conformal prediction for crowd navigation," *arXiv preprint arXiv:2502.06221*, 2025.

[31] K. J. Strawn, N. Ayanian, and L. Lindemann, "Conformal predictive safety filter for rl controllers in dynamic environments," *IEEE Robotics and Automation Letters*, vol. 8, no. 11, pp. 7833–7840, 2023.

[32] H. Huang, S. Sharma, A. Loquercio, A. Angelopoulos, K. Goldberg, and J. Malik, "Conformal policy learning for sensorimotor control under distribution shifts," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 16 285–16 291.

[33] J. Lekeufack, A. N. Angelopoulos, A. Bajcsy, M. I. Jordan, and J. Malik, "Conformal decision theory: Safe autonomous decisions from imperfect predictions," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 11 668–11 675.

[34] R. J. Tibshirani, R. Foygel Barber, E. Candes, and A. Ramdas, "Conformal prediction under covariate shift," *Advances in neural information processing systems*, vol. 32, 2019.

[35] A. Dixit, L. Lindemann, S. X. Wei, M. Cleaveland, G. J. Pappas, and J. W. Burdick, "Adaptive conformal prediction for motion planning among dynamic agents," in *Learning for Dynamics and Control Conference*. PMLR, 2023, pp. 300–314.

[36] J. Yao, X. Zhang, Y. Xia, Z. Wang, A. K. Roy-Chowdhury, and J. Li, "Towards generalizable safety in crowd navigation via conformal uncertainty handling," *arXiv preprint arXiv:2508.05634*, 2025.

[37] O. Bastani, "Safe reinforcement learning with nonlinear dynamics via model predictive shielding," in *2021 American control conference (ACC)*. IEEE, 2021, pp. 3488–3494.

[38] K. P. Wabersich, L. Hewing, A. Carron, and M. N. Zeilinger, "Probabilistic model predictive safety certification for learning-based control," *IEEE Transactions on Automatic Control*, vol. 67, no. 1, pp. 176–188, 2021.

[39] Z. Han, Y. Wu, T. Li, L. Zhang, L. Pei, L. Xu, C. Li, C. Ma, C. Xu, S. Shen *et al.*, "An efficient spatial-temporal trajectory planner for autonomous vehicles in unstructured environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 2, pp. 1797–1814, 2023.

[40] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Aggressive driving with model predictive path integral control," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1433–1440.

[41] R. Khanam and M. Hussain, "Yolov11: An overview of the key architectural enhancements," *arXiv preprint arXiv:2410.17725*, 2024.

[42] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," in *European conference on computer vision*. Springer, 2022, pp. 1–21.