Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Uncertainty-Guided Belief World Models for Long-Horizon UAV Navigation under Partial Observability and Visual Degradation

Dhayaa Khudher

dhayaa.khudher@uobasrah.edu.iq

University of Basrah

Research Article

Keywords: UAV navigation, latent world models, partial observability, posterior confidence, predictive uncertainty, visual degradation, model-based reinforcement learning, selective replanning

Posted Date: November 5th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10749144/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Uncertainty-Guided Belief World Models for Long-Horizon UAV Navigation under Partial Observability and Visual Degradation

Dhayaa R. Khudher[1*]

[1*]Department of Computer Engineering, University of Basrah, Basrah, Iraq.

Corresponding author(s). E-mail(s): dhayaa.khudher@uobasrah.edu.iq;

Abstract

Long-horizon UAV navigation under partial observability and visual degradation requires reliable decisions from uncertain current beliefs and increasingly uncertain imagined futures. We propose the Uncertainty-Guided Belief World Model (UG-BWM), which exposes posterior covariance to the actor and critic, propagates ensemble-based predictive uncertainty through latent rollouts, penalizes uncertain and collision-prone imagined transitions, and triggers candidate-sequence replanning through a hysteretic uncertainty rule. We evaluate UG-BWM in Gazebo Harmonic with PX4 SITL across six locked-test conditions, three route lengths, four visual-degradation types, and unseen structural shifts. UG-BWM achieves a success rate of 0.63, a collision rate of 0.15, and an SPL of 0.62, compared with 0.54, 0.20, and 0.54 for DreamerV3-CU. Controlled tests show that posterior confidence, the combined predictive uncertainty score, and the state-dependent imagined objective each contribute to performance. Predictive uncertainty remains positively associated with rollout error beyond the training horizon. During deployment, uncertainty-triggered replanning matches always-plan performance while activating the planner on 18% of control steps, reducing planner use by 82%. The results support uncertainty-explicit belief modeling and selective planning for UAV navigation under degraded sensing.

Keywords: UAV navigation, latent world models, partial observability, posterior confidence, predictive uncertainty, visual degradation, model-based reinforcement learning, selective replanning

1 Introduction

Most civilian and military applications of autonomous UAVs depend on onboard sensing and state-estimation systems for navigation [1]. Real-world UAV navigation is complicated by visual degradation, including fog, low illumination, motion blur, and occlusion, as well as by partial observability [2, 3]. The problem becomes more severe in long-horizon settings, where navigation errors accumulate and decisions must be revised using an incomplete representation of the environment [4, 5]. Goal-directed UAV navigation becomes more difficult when the vehicle must operate in unknown environments with incomplete or noisy visual observations [6].

Vision-based reinforcement learning enables navigation policies to be learned directly from onboard camera observations [7, 8]. Recent methods address cluttered environments [9], unknown geometry [10], temporal dependencies [11], and perceptual uncertainty [12]. These methods use model-free control, recurrent memory, temporal attention, and safety mechanisms [13]. They can learn reactive or history-dependent policies. However, they do not explicitly predict how the UAV belief and the environment will change under alternative future actions. Their recurrent components encode observation history without directly indicating the confidence of the current inferred state.

World models learn compact latent representations that predict how the environment changes under the actions of the agent [14–16]. Predicted trajectories are used for online planning or policy optimization through imagined rollouts [17]. World models have also been applied to visual representation learning for autonomous drones and navigation control for robotic agents [18, 19]. Their effectiveness depends on the quality of the inferred current latent state and the accuracy of recursively predicted future states. When a policy optimizes actions using unreliable imagined trajectories, prediction errors can propagate through the rollout and lead to poor decisions.

Existing uncertainty-aware world models reduce the influence of unreliable predictions during learning or control [20, 21]. However, they generally do not distinguish confidence in the current observation-conditioned belief from uncertainty accumulated through recursive latent prediction. This distinction is important under visual degradation because uncertainty may already be present before imagination begins and may increase as the rollout progresses.

This work proposes UG-BWM for long-horizon UAV navigation under partial observability and visual degradation. UG-BWM separates confidence in the current posterior belief from uncertainty accumulated through imagined future transitions. It exposes posterior confidence to the actor and critic, estimates predictive uncertainty through an ensemble latent world model, and uses this uncertainty to guide imagined policy learning and selectively activate deployment-time planning.

The main contributions of this work are:

- An uncertainty-explicit belief interface that supplies the actor and critic with both the observation-conditioned latent estimate and its posterior confidence.
- An ensemble predictive prior that propagates latent beliefs and combines ensemble disagreement with conditional predictive variance to estimate predictive uncertainty across imagined horizons.

- An uncertainty-guided imagined actor–critic objective that reduces the influence of unreliable latent transitions and conservatively accounts for predicted collision risk.
- A selective replanning mechanism that uses predictive uncertainty to activate additional planning only at states where the nominal policy rollout becomes unreliable.

2 Related Work

2.1 UAV Navigation under Partial Observability and Perceptual Degradation

Recent literature groups autonomous UAV navigation into classical planning approaches [22, 23], learning-based navigation [24, 25], models designed for partial observability [26], and strategies developed for degraded sensing conditions [27]. Common classical planning approaches include artificial potential fields [28, 29], sampling-based algorithms [4], or combinations of both [30]. Such methods depend on accurate vehicle states, obstacle representations, and map quality.

In learning-based navigation, the policy is trained from sensor observations rather than relying on an explicit planner [31]. Model-free reinforcement-learning algorithms such as PPO [32], SAC [33], and TD3 [34] are used for obstacle avoidance and navigation in constrained environments by mapping visual and motion inputs directly to continuous actions. The methods differ in their observation encoders, policy architectures, reward formulations, and safety mechanisms [35]. Most learn actions from the current observation or from compact representations of recent observations.

Partial observability has led to greater use of recurrent memory [36], temporal attention [37], and spatiotemporal feature modeling [38]. Related studies use spatiotemporal Transformers, perception-enhanced memory with temporal-region selection, and attention-based temporal fusion [39]. Such models retain information that is not available in the current frame. The retained information is usually represented as a hidden state or temporal feature. In the reviewed methods, however, no explicit indication is provided of how reliable that representation remains under degraded or missing observations.

Recent methods address UAV navigation under perception noise [12] and under degraded visibility with limited sensor capabilities [40]. Some approaches focus on motion estimation from degraded images [41]. Other studies modify the navigation policy or introduce safety mechanisms [42]. Prior work on reduced visibility [27], image blur [41], uncertain optical flow [43], and limited sensor capabilities [40] shows that visual quality directly affects navigation decisions.

2.2 Latent World Models and Model-Based Reinforcement Learning

World models use learned latent dynamics to optimize policies without requiring new environment interaction at every update [16]. The main methods differ in the representation of the latent state and in how the learned world model supports control [44]. PlaNet optimizes action sequences online in a latent space learned from pixel

observations [15], whereas Dreamer uses imagined trajectories to train the actor and critic networks [16]. DreamerV2 introduced discrete latent variables, while DreamerV3 retained the imagined actor–critic architecture and applied it to a broader range of control tasks [44, 45].

Other model-based reinforcement learning (MBRL) methods differ in their representation of the dynamics and their use of model predictions. PETS propagates probabilistic dynamics ensembles and evaluates candidate actions through trajectory sampling [46]. MBPO uses short synthetic rollouts to reduce the accumulation of model error [47]. TD-MPC combines learned latent dynamics, temporal-difference learning, and model-predictive control [17]. These approaches differ in three related factors. Rollout length influences exposure to compounding prediction errors [47]. Probabilistic dynamics models represent uncertainty in learned predictions [48], while online action selection determines whether model predictions are used directly for control or mainly for policy training [17].

Robotic applications use world models for different purposes. Zhao et al. [49] applied a contrastive learning objective within a world model to learn visual representations for autonomous drone navigation. Shanks et al. [18] developed DreamerNav, which uses imagined control for navigation in dynamic indoor environments. Martinez-Baselga et al. [50] introduced RUMOR, which learns an environment model for navigation in dynamic scenes. Tommaselli et al. [51] developed LeCropFollow, which uses latent-space planning for navigation in unstructured crop fields. Fujii and Murata [52] used a temporally hierarchical world model for real-world robot control. Boiteau et al. [40] studied MBRL decision making for UAVs operating under degraded visibility.

Two design issues are relevant to the present work. First, stochastic latent models may still provide the actor and critic with only a sampled or point representation, without direct access to posterior confidence [44, 45]. Second, existing evaluations usually report prediction accuracy or task performance at selected horizons [17]. Few studies test whether predictive uncertainty continues to rank realized error as the imagined rollout becomes longer.

2.3 Uncertainty-Aware Model-Based Reinforcement Learning

In MBRL, the uncertainty signal can regulate the impact of model-generated transitions on policy learning. It can reduce the contribution of unreliable predictions [53], or constrain the policy [54]. For example, MOPO penalizes the reward assigned to imagined transitions according to model uncertainty. A related conservative method, COMBO, combines model-based policy optimization with conservative value estimation to regulate the value-learning process [55].

Other approaches use uncertainty through trajectory correction [20], and reachability-based safety certificates for UAV flight [56]. Work in autonomous driving also uses epistemic uncertainty for safe reinforcement learning [57] and propagates uncertainty through hierarchical planning [58]. In each case, uncertain predictions have less influence on control than predictions considered reliable.

Uncertainty may also help determine when additional computation is needed during deployment. Most of the reviewed methods apply the uncertainty signal continuously as a reward penalty, policy constraint, trajectory-correction signal, or safety

measure. The reviewed studies do not examine whether uncertainty can identify states at which online planning is more useful than acting directly from the learned policy.

These functions are generally investigated independently. The reviewed methods do not jointly expose posterior confidence to the policy, propagate predictive uncertainty through imagined beliefs, evaluate its association with realized error across rollout depths, and use it to determine when deployment-time replanning should be activated. The proposed UG-BWM integrates these functions within one framework.

3 Problem Formulation

This work considers long-horizon UAV navigation under partial observability, visual degradation, and dynamic obstacles. The formulation distinguishes uncertainty in the current observation-conditioned belief from uncertainty in future latent predictions because they arise at different stages of the decision process.

3.1 Partially Observable Navigation and Mission Objective

The navigation task is modeled as a partially observable Markov decision process (POMDP):

$$\begin{aligned} \mathcal{M} &= (\mathcal{X}, \mathcal{A}, \mathcal{O}, P, \Omega, R, C, \gamma), \\ x_{t+1} &\sim P(x_{t+1} \mid x_t, a_t), \qquad o_t \sim \Omega(o_t \mid x_t), \\ r_t &= R(x_t, a_t), \qquad c_t = C(x_t, a_t), \quad c_t \in \{0, 1\}. \end{aligned} \tag{1}$$

Here, $\mathcal{X}$, $\mathcal{A}$, and $\mathcal{O}$ denote the hidden-state, continuous-action, and onboard-observation spaces; P and Ω are the transition and observation models; R is the task reward; C is the binary safety cost; and $\gamma \in [0, 1)$ is the discount factor. The hidden state includes the UAV motion state, environment geometry, obstacle motion, and visual conditions. The observed event $c_t = 1$ denotes a collision or permitted-flight-boundary violation and is distinct from a collision probability predicted by the learned world model.

Because x_t is hidden, the policy acts from the interaction history $\mathcal{H}_t = (o_0, a_0, \ldots, a_{t-1}, o_t)$. The nominal mission objective is

$$J(\pi) = \mathbb{E}_\pi \left[\sum_{t=0}^{T-1} \gamma^t r_t \right], \tag{2}$$

where T is the episode duration. Since a mission may require more control steps than the learned prediction horizon, long-horizon navigation is represented as repeated posterior-update and finite-horizon decision cycles.

The safety event c_t supplies collision and permitted-flight-boundary labels for world-model learning and safety evaluation. The formulation does not impose a hard safety constraint. Instead, UG-BWM incorporates predicted collision risk into the penalized imagined return.

3.2 Onboard Observations and Uncertainty-Explicit Belief

The onboard observation and high-level body-frame command are:

$$
\begin{aligned}
o_t &= \{I_t, \eta_t, g_t, m_t\}, \\
a_t &= [v^b_{x,t}, v^b_{y,t}, v^b_{z,t}, \dot{\psi}_t]^\top \in \mathcal{A}.
\end{aligned}
\tag{3}
$$

I_t is the RGB image, η_t is the PX4 motion estimate, g_t is the relative goal representation, and m_t indicates camera availability. Degraded but available images use $m_t = 1$, whereas dropped or unavailable frames use $m_t = 0$. Simulator ground truth, global obstacle states, and future information are excluded from the policy input.

A single observation does not generally determine the hidden navigation state. The policy therefore acts from an observation-conditioned belief:

$$
\begin{aligned}
\mathcal{B}^q_t(x) &= p(x_t = x \mid \mathcal{H}_t), \\
\mathcal{B}^q_t &= \mathfrak{B}(\mathcal{B}^q_{t-1}, a_{t-1}, o_t), \\
a_t &\sim \pi(a_t \mid \mathcal{B}^q_t).
\end{aligned}
\tag{4}
$$

The superscript q denotes conditioning on observations available through time t. For the nonlinear latent dynamics and high-dimensional visual observation model considered here, exact posterior inference is generally intractable. A learned approximate posterior is therefore required to represent both the inferred latent state and its posterior uncertainty after incorporating the current observation and recurrent history [15, 16].

3.3 Finite-Horizon Imagination and Uncertainty-Aware Decision Making

Starting from the current posterior belief b^q_t, an imagined action sequence $a_{t:t+H-1} = (a_t, \ldots, a_{t+H-1})$ generates predictive beliefs $\hat{b}^p_{t+1}, \ldots, \hat{b}^p_{t+H}$ without conditioning on the future observations corresponding to those states. Each imagined transition is:

$$
\begin{aligned}
\mathcal{W}\left(\hat{b}^p_{t+k}, a_{t+k}\right) = \Big(&\hat{b}^p_{t+k+1}, \hat{r}_{t+k}, \\
&\hat{c}_{t+k}, U_{t+k}\Big), \quad k = 0, \ldots, H-1.
\end{aligned}
\tag{5}
$$

with $\hat{b}^p_t = b^q_t$. Here, $\hat{r}_{t+k}$ is the predicted reward, $\hat{c}_{t+k} \in [0, 1]$ is the predicted collision probability, and U_{t+k} is the predictive uncertainty score associated with the imagined transition.

Posterior uncertainty concerns the current belief after conditioning on the available observation history. Predictive uncertainty concerns future latent beliefs generated before their corresponding observations become available. Posterior correction at the next interaction step therefore does not eliminate the need to assess the reliability of the rollout used for the current decision.

The resulting decision problem requires representation of the inferred current state, confidence in the observation-conditioned belief, and uncertainty associated with recursively imagined future states.

4 Proposed Method

Figure 1 summarizes the two computational paths of UG-BWM. The posterior path assimilates observations available at the current control step and forms an uncertainty-explicit belief for the actor and critic. Starting from this belief, the predictive path propagates finite-horizon latent trajectories through an ensemble world model and estimates reward, collision risk, and predictive uncertainty for imagined learning and selective deployment-time planning.

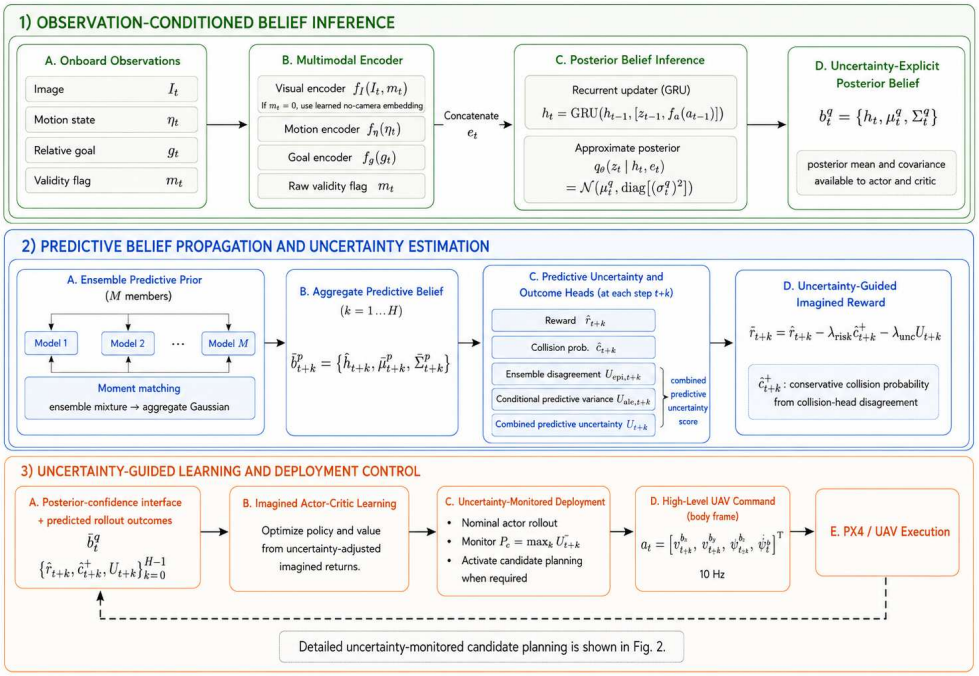


Fig. 1 UG-BWM architecture for uncertainty-explicit belief inference, imagined learning, and selective deployment-time planning.

4.1 Multimodal Encoding and Uncertainty-Explicit Posterior Belief

Separate encoders process the RGB image, PX4 motion estimate, and relative goal. Their outputs are concatenated with the camera-validity flag:

$$e_t = [f_I(I_t, m_t), f_\eta(\eta_t), f_g(g_t), m_t]\,. \tag{6}$$

For $m_t = 1$, f_I returns the CNN feature of the available image, including visually degraded frames. For $m_t = 0$, the visual feature is replaced by a learned

no-camera embedding; the zero-valued image tensor only preserves the fixed input shape. Including m_t in e_t distinguishes unavailable visual evidence from ordinary image content.

Temporal history, posterior inference, and the uncertainty-explicit belief representation are defined jointly as:

$$\begin{aligned} h_t &= \mathrm{GRU}_\phi\left(h_{t-1}, [z_{t-1}, f_a(a_{t-1})]\right), \\ q_\phi(z_t \mid h_t, e_t) &= \mathcal{N}\big(\mu_t^q, \mathrm{diag}[(\sigma_t^q)^2]\big), \\ z_t &= \mu_t^q + \sigma_t^q \odot \epsilon_t, \qquad \epsilon_t \sim \mathcal{N}(0, I), \\ \Sigma_t^q &= \mathrm{diag}[(\sigma_t^q)^2], \qquad b_t^q = \{h_t, \mu_t^q, \Sigma_t^q\}, \\ \bar{b}_t^q &= [h_t, \mu_t^q, \log \mathrm{diag}(\Sigma_t^q)]. \end{aligned} \tag{7}$$

The actor and critic receive $\bar{b}_t^q$; hence, posterior log-covariance is explicitly available to the decision networks rather than being represented only through a sampled latent state. During real interaction, the decision networks receive the observation-conditioned representation $\bar{b}_t^q$. During imagined learning, the same moment-explicit interface is retained through the ensemble-propagated representation $\bar{b}_{t+k}^p$.

4.2 Ensemble Predictive Belief and Uncertainty Estimates

Each imagined rollout begins from the latest posterior belief and propagates future states through M independently parameterized predictive prior heads. For action a_{t+k},

$$\begin{aligned} \hat{h}_{t+k+1} &= \mathrm{GRU}_\phi\left(\hat{h}_{t+k}, [\hat{z}_{t+k}, f_a(a_{t+k})]\right), \\ p_{\theta_m}(z_{t+k+1} \mid \hat{h}_{t+k+1}) &= \mathcal{N}\Big(\mu_{t+k+1}^{p,(m)}, \mathrm{diag}[(\sigma_{t+k+1}^{p,(m)})^2]\Big). \end{aligned} \tag{8}$$

The ensemble mixture is approximated by matching its first two moments:

$$\begin{aligned} \bar{\mu}_{t+k+1}^p &= \frac{1}{M}\sum_{m=1}^{M} \mu_{t+k+1}^{p,(m)}, \\ (\bar{\sigma}_{t+k+1}^p)^2 &= \frac{1}{M}\sum_{m=1}^{M}\left[(\sigma_{t+k+1}^{p,(m)})^2 + \left(\mu_{t+k+1}^{p,(m)} - \bar{\mu}_{t+k+1}^p\right)^2\right], \\ \bar{\Sigma}_{t+k+1}^p &= \mathrm{diag}[(\bar{\sigma}_{t+k+1}^p)^2], \\ \hat{b}_{t+k+1}^p &= \{\hat{h}_{t+k+1}, \bar{\mu}_{t+k+1}^p, \bar{\Sigma}_{t+k+1}^p\}, \\ \bar{b}_{t+k+1}^p &= [\hat{h}_{t+k+1}, \bar{\mu}_{t+k+1}^p, \log \mathrm{diag}(\bar{\Sigma}_{t+k+1}^p)]. \end{aligned} \tag{9}$$

This first-two-moment approximation propagates one aggregate Gaussian belief per imagined action sequence and therefore does not preserve separate predictive modes. A reparameterized aggregate sample is used during imagined training, whereas deterministic evaluation uses the aggregate mean.

Reward and collision predictions retain the member-specific latent moments:

$$
\begin{aligned}
\bar{b}^{p,(m)}_{t+k+1} &= [\hat{h}_{t+k+1}, \mu^{p,(m)}_{t+k+1}, \log(\sigma^{p,(m)}_{t+k+1})^2], \\
\hat{r}^{(m)}_{t+k} &= r_{\theta_m}(\bar{b}^{p,(m)}_{t+k+1}), \\
\hat{c}^{(m)}_{t+k} &= \mathrm{sigmoid}\Big(c_{\theta_m}(\bar{b}^{p,(m)}_{t+k+1})\Big), \\
\hat{r}_{t+k} &= \frac{1}{M}\sum_{m=1}^{M} \hat{r}^{(m)}_{t+k}.
\end{aligned}
\tag{10}
$$

$t+k$ denotes the transition initiated by a_{t+k}, so the post-update predictive belief estimates the corresponding reward and collision outcome. The latent dimension is denoted by d. Ensemble disagreement and average conditional predictive variance are:

$$
\begin{aligned}
U_{\mathrm{epi},t+k} &= \frac{1}{Md}\sum_{m=1}^{M}\sum_{j=1}^{d}\left(\mu^{p,(m)}_{t+k+1,j} - \bar{\mu}^{p}_{t+k+1,j}\right)^2, \\
U_{\mathrm{ale},t+k} &= \frac{1}{Md}\sum_{m=1}^{M}\sum_{j=1}^{d}\left(\sigma^{p,(m)}_{t+k+1,j}\right)^2.
\end{aligned}
\tag{11}
$$

The labels epi and ale follow conventional terminology. In this work, however, U_{epi} denotes ensemble disagreement and U_{ale} denotes the ensemble-averaged conditional predictive variance. They are treated as operational uncertainty indicators rather than uniquely identifiable epistemic and aleatoric sources. Each component is normalized by a fixed calibration reference Q_s, defined as its empirical 95th percentile over nominal rollouts from the calibration manifest. The references are recomputed for the selected checkpoint and frozen during locked testing; Section 5.8 specifies the calibration protocol. The normalized components and combined predictive uncertainty score are:

$$
\begin{aligned}
\bar{U}_{s,t+k} &= \mathrm{clip}\left(\frac{U_{s,t+k}}{Q_s + \varepsilon}, 0, 1\right), \quad s \in \{\mathrm{epi}, \mathrm{ale}\}, \\
U_{t+k} &= \frac{\bar{U}_{\mathrm{epi},t+k} + \alpha \bar{U}_{\mathrm{ale},t+k}}{1+\alpha}.
\end{aligned}
\tag{12}
$$

Collision probability remains separate from latent predictive uncertainty. The ensemble mean, collision-head disagreement, and conservative estimate are:

$$
\begin{aligned}
\hat{c}_{t+k} &= \frac{1}{M}\sum_{m=1}^{M}\hat{c}^{(m)}_{t+k}, \\
U^{c}_{t+k} &= \frac{1}{M}\sum_{m=1}^{M}\left(\hat{c}^{(m)}_{t+k} - \hat{c}_{t+k}\right)^2, \\
\hat{c}^{+}_{t+k} &= \mathrm{clip}\left(\hat{c}_{t+k} + \beta_c\sqrt{U^{c}_{t+k}}, 0, 1\right).
\end{aligned}
\tag{13}
$$

The coefficients α and β_c are selected through the model-selection protocol in Section 5.8 and remain fixed during locked testing.

4.3 World-Model Learning

Training uses trajectory sequences sampled from the replay buffer. The posterior is inferred from real observations, while each predictive prior estimates the subsequent latent state from recurrent history and the executed action. A shared decoder reconstructs the image, PX4 estimate, and relative goal from (h_t, z_t). With $q_{t+1} = q_\phi(z_{t+1} \mid h_{t+1}, e_{t+1})$, $p_{t+1}^{(m)} = p_{\theta_m}(z_{t+1} \mid h_{t+1})$, and $\mathrm{sg}[\cdot]$ denoting stop gradient, the loss terms are

$$\begin{aligned}
\mathcal{L}_{\mathrm{obs}} &= \lambda_I \|I_t - \hat{I}_t\|_1 + \lambda_\eta \|\eta_t - \hat{\eta}_t\|_2^2 + \lambda_g \|g_t - \hat{g}_t\|_2^2, \\
\mathcal{L}_{\mathrm{KL}} &= \frac{1}{M} \sum_{m=1}^{M} \Big[\xi D_{\mathrm{KL}}(\mathrm{sg}[q_{t+1}] \| p_{t+1}^{(m)}) \\
&\qquad + (1-\xi) D_{\mathrm{KL}}(q_{t+1} \| \, \mathrm{sg}[p_{t+1}^{(m)}]) \Big], \\
\mathcal{L}_r &= \frac{1}{M} \sum_{m=1}^{M} (r_t - \hat{r}_t^{(m)})^2, \\
\mathcal{L}_c &= -\frac{1}{M} \sum_{m=1}^{M} \Big[c_t \log \hat{c}_t^{(m)} + (1-c_t) \log(1-\hat{c}_t^{(m)}) \Big].
\end{aligned} \tag{14}$$

The complete world-model objective is

$$\mathcal{L}_{\mathrm{WM}} = \mathcal{L}_{\mathrm{obs}} + \lambda_{\mathrm{KL}} \mathcal{L}_{\mathrm{KL}} + \lambda_r \mathcal{L}_r + \lambda_c \mathcal{L}_c. \tag{15}$$

The predictive prior, reward, and collision heads are independently initialized following deep-ensemble training [48]. To diversify the training data assigned to individual ensemble members, we apply sequence-level Bernoulli bootstrap masks, following the bootstrapped dynamics-ensemble principle [46]. Each mask $w_{i,m} \sim \mathrm{Bernoulli}(p_{\mathrm{boot}})$ is drawn once for sampled sequence i and ensemble member m, held fixed over the complete sequence, and applied to the member-specific KL, reward, and collision terms.

4.4 Uncertainty-Guided Imagined Actor–Critic Learning

Imagined rollouts begin from posterior beliefs sampled from the replay buffer. The actor and critic receive the aggregate predictive representation $\bar{b}_{t+k}^p$ in Eq. (9), which has the same moment-explicit interface as $\bar{b}_t^q$ but obtains its covariance through ensemble propagation. The bounded stochastic action is:

$$\begin{aligned}
a_{t+k} &= a_{\max} \odot \tanh\big(\mu_\rho(\bar{b}_{t+k}^p) + \sigma_\rho(\bar{b}_{t+k}^p) \odot \epsilon_{t+k}\big), \\
\epsilon_{t+k} &\sim \mathcal{N}(0, I).
\end{aligned} \tag{16}$$

For each imagined transition, UG-BWM combines predicted reward with conservative collision risk and predictive uncertainty:

$$\tilde{r}_{t+k} = \hat{r}_{t+k} - \lambda_{\mathrm{risk}} \hat{c}_{t+k}^{+} - \lambda_{\mathrm{unc}} U_{t+k}. \tag{17}$$

Algorithm 1 Compact training cycle for UG-BWM

Initialize replay buffer $\mathcal{D}$, world model, predictive ensemble, actor π_ρ, critic V_ν, and target critic $V_{\bar{\nu}}$
Collect initial trajectories and store them in $\mathcal{D}$
for each training iteration **do**
    Execute the stochastic actor and append transitions to $\mathcal{D}$
    Sample trajectory segments, infer posterior beliefs, and draw sequence-level ensemble bootstrap masks
    Update the world model by minimizing $\mathcal{L}_{\mathrm{WM}}$
    if the calibration interval K_{cal} is reached **then**
        Recompute Q_{epi} and Q_{ale} from nominal rollouts on the fixed calibration manifest
    end if
    Freeze the world model and generate H-step imagined rollouts from replay-buffer posterior states
    Update the actor and critic using the adjusted returns in Eqs. (17) and (18)
    Update the target critic by exponential averaging
end for

The first penalty addresses predicted collision outcomes, while the second reduces the contribution of imagined transitions with high predictive uncertainty. Returns and actor–critic objectives are:

$$\begin{aligned} G_{t+H} &= V_{\bar{\nu}}(\bar{b}^p_{t+H}), \\ G_{t+k} &= \tilde{r}_{t+k} + \gamma G_{t+k+1}, \quad k = H-1, \dots, 0, \\ \mathcal{L}_V &= \mathbb{E}\left[\frac{1}{H}\sum_{k=0}^{H-1}\left(V_\nu(\bar{b}^p_{t+k}) - \mathrm{sg}[G_{t+k}]\right)^2\right], \\ \mathcal{L}_\pi &= -\mathbb{E}[G_t]. \end{aligned} \tag{18}$$

World-model parameters are fixed during actor and critic updates, and the target critic $V_{\bar{\nu}}$ is updated by exponential averaging. Algorithm 1 summarizes the resulting training cycle; deployment-time candidate planning is disabled during data collection and training.

4.5 Uncertainty-Monitored Deployment and Selective Latent-Space Planning

Figure 2 summarizes deployment-time action selection. From the current posterior belief, the deterministic actor generates a nominal rollout over a monitoring horizon H_{mon}. Let U^{π}_{t+k} denote Eq. (12) evaluated along this rollout; the superscript identifies the rollout source rather than a separate uncertainty measure.

The monitoring statistic and hysteretic activation rule are:

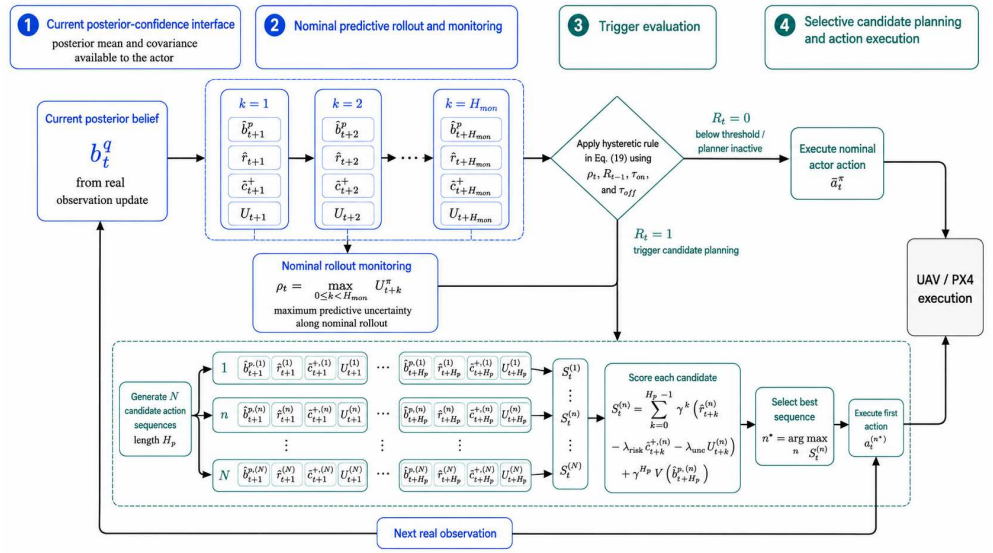


Fig. 2 UG-BWM deployment-time action selection. A hysteretic predictive-uncertainty trigger switches between nominal actor control and actor-centered latent-space planning.

$$\rho_t = \max_{0 \le k < H_{\text{mon}}} U_{t+k}^{\pi},$$
$$\mathcal{R}_t = \begin{cases} 1, & \rho_t > \tau_{\text{on}}, \\ 0, & \rho_t < \tau_{\text{off}}, \\ \mathcal{R}_{t-1}, & \text{otherwise}, \end{cases} \qquad \tau_{\text{off}} < \tau_{\text{on}}. \tag{19}$$

When $\mathcal{R}_t = 0$, the deterministic actor mean is executed. When $\mathcal{R}_t = 1$, the planner samples N actor-centered sequences of horizon H_p:

$$a_{t+k}^{(n)} = a_{\max} \odot \tanh\left(\mu_\rho(\bar{b}_{t+k}^{p,(n)}) + \epsilon_{t+k}^{(n)}\right),$$
$$\epsilon_{t+k}^{(n)} \sim \mathcal{N}(0, \Sigma_{\text{plan}}). \tag{20}$$

All candidates start from the same posterior belief, and one uses zero perturbation so that the nominal actor sequence is retained. Candidate scoring and receding-horizon selection are:

$$S_t^{(n)} = \sum_{k=0}^{H_p-1} \gamma^k \tilde{r}_{t+k}^{(n)} + \gamma^{H_p} V_\nu(\bar{b}_{t+H_p}^{p,(n)}),$$
$$n^* = \arg\max_{1 \le n \le N} S_t^{(n)}, \qquad a_t = a_t^{(n^*)}. \tag{21}$$

Here, $\tilde{r}_{t+k}^{(n)}$ follows Eq. (17). Only the first selected action is sent to PX4; the next real observation corrects the posterior before monitoring and action selection

are repeated. All learned parameters and uncertainty-normalization constants remain fixed during deployment.

The posterior-confidence interface, predictive-uncertainty components, imagined objective, and replanning trigger are evaluated through the controlled tests in Section 5.5. Their results are reported in Sections 6.4–6.7. In the trigger comparison, the candidate planner is held fixed so that only the activation rule changes.

5 Experimental Setup and Evaluation Protocol

The evaluation compares UG-BWM with TD3, recurrent PPO, DreamerV3, and DreamerV3-CU, where CU denotes conservative uncertainty-aware imagination. DreamerV3-CU is a controlled extension of DreamerV3 that adds ensemble predictive uncertainty and an uncertainty-penalized imagined objective. Controlled tests separately examine posterior confidence, predictive uncertainty, imagined objective guidance, and selective replanning. All methods use five independent seeds and paired evaluation episodes, with disjoint training, calibration, model-selection, and locked-test manifests.

5.1 Simulation Platform and System Integration

Experiments use Gazebo Harmonic, PX4 software-in-the-loop simulation, ROS 2, and the Micro XRCE-DDS bridge. Gazebo provides the environment geometry, vehicle dynamics, RGB observations, moving obstacles, controlled visual degradation, collision events, and evaluation-only ground truth. PX4 performs onboard state estimation, low-level stabilization, and actuator control. ROS 2 synchronizes observations, estimator outputs, and high-level commands with the PyTorch navigation framework.

At each control step, the policy receives only the onboard variables defined in Eq. (3). The relative goal is calculated from PX4 position and heading estimates. Simulator poses, global obstacle states, degradation labels, future observations, and reference paths are excluded from the policy input, posterior inference, imagined rollouts, predictive uncertainty, and candidate planner.

5.2 Task Specification and Data Separation

The evaluation uses six training layouts, three validation layouts, two locked Test-ID layouts, and two locked Test-OOD layouts. Test-ID layouts retain the structural family represented during training but use disjoint start–goal pairs and obstacle realizations. Test-OOD layouts introduce unseen connectivity, passage dimensions, obstacle arrangements, and decision points.

Table 1 summarizes the observation interface, route classes, data partitions, and episode allocation. No start–goal pair, obstacle trajectory, degradation schedule, or simulator seed is shared across the training, validation, and locked-test partitions.

Reference length L_i^* is the shortest feasible path through the static traversable geometry. An offline planner with complete map access is used only to verify feasibility, assign route categories, and calculate SPL. Its map, waypoints, and actions are not provided to any learned method.

Table 1 Task specification, data separation, and route protocol.

Category	Setting	Protocol
Observation	RGB, PX4 motion estimate, relative goal, and camera-validity flag	All learned methods receive the same onboard information. Simulator state and global obstacle information are excluded.
Action	Body-frame velocity and yaw-rate command	$a_t = [v^b_{x,t}, v^b_{y,t}, v^b_{z,t}, \dot{\psi}_t]^\top$ at a 10-Hz high-level control rate.
Layout split	6 training; 3 validation; 2 Test-ID; 2 Test-OOD	Validation contains separate calibration and model-selection manifests. Test layouts remain locked until final evaluation.
Short route	$10 \le L^*_i < 20$ m	Approximately 50–100 high-level decision steps.
Medium route	$20 \le L^*_i < 40$ m	Approximately 100–200 high-level decision steps.
Long route	$40 \le L^*_i \le 70$ m	Approximately 200–350 decision steps, substantially exceeding the $H = 15$ imagination horizon.
Primary manifest	20 routes per route class, condition, and seed	Each condition contains 60 episodes per seed and 300 episodes per method across five trained seeds.
Severity manifest	Paired clear, mild, moderate, and severe trials	A fixed route subset is replayed across degradation levels to construct Fig. 5.
Stress manifest	Test-OOD geometry with unseen compound degradation	Contains balanced short, medium, and long routes and is used for the route-horizon and combined-shift analyses.

The route-dependent episode limit is

$$T_{\max,i} = T_{\text{margin}} + \kappa_{\text{time}} \frac{L^*_i}{v_{\text{ref}}}, \tag{22}$$

with common values of T_{margin}, κ_{time}, and v_{ref} for all methods. Long-horizon navigation is therefore evaluated through repeated finite-horizon imagination and posterior correction rather than an open-loop prediction over the full route.

5.3 Visual Degradation and Locked-Test Conditions

Visual disturbances are applied to the RGB stream before policy preprocessing. Degraded but available images retain $m_t = 1$, whereas unavailable frames use the missing-camera representation and $m_t = 0$. Disturbance schedules specify onset, duration, severity, and recovery independently of the policy state.

Table 2 defines the six primary locked-test conditions. Each disturbed episode begins with a nominal observation interval so that the response to disturbance onset can be evaluated without changing the initial-state distribution.

Table 2 Locked-test conditions and controlled disturbance factors.

ID	Condition	Configuration	Controlled variables
C1	Clear ID	Test-ID layouts under nominal sensing and standard obstacle density	Nominal illumination, no synthetic fog or occlusion, and standard obstacle trajectories.
C2	Severe visual degradation	Test-ID layouts with one severe visual disturbance	Fog, low illumination, motion blur, or partial occlusion, balanced across episodes.
C3	Dense dynamic obstacles	Test-ID layouts under nominal vision with increased moving-obstacle complexity	Obstacle density, speed, crossing direction, and motion diversity.
C4	Unseen degradation combination	Test-ID layouts with compound visual disturbances excluded from training	At least two degradation factors with unseen severity combinations or temporal schedules.
C5	Unseen geometry	Test-OOD layouts under nominal sensing	Unseen connectivity, passage width, obstacle arrangement, and decision-point placement.
C6	Camera blackout	Test-ID layouts with sustained unavailable-image intervals	Blackout onset, duration, recovery, and obstacle motion during sensor loss.

The severity-sweep manifest varies fog, illumination, blur, and occlusion independently over clear, mild, moderate, and severe settings. The four categories follow the clear-to-severe parameter ordering reported in Table 4: fog visibility and illumination decrease, whereas blur-kernel size and occluded-image fraction increase. The same routes, obstacle seeds, and disturbance onset times are reused across severity levels.

The dense-dynamic-obstacle condition (C3) isolates obstacle-motion difficulty from visual degradation. The unseen-degradation condition (C4) evaluates visual combinations and disturbance schedules excluded from training. The unseen-geometry condition (C5) isolates structural novelty under nominal sensing. The stress manifest combines C4-type visual disturbances with Test-OOD layouts to evaluate simultaneous perceptual and structural shift.

5.4 Compared Methods and Fairness Controls

UG-BWM is compared with TD3 [34], recurrent PPO [32], DreamerV3 [44], and the controlled DreamerV3-CU baseline. Table 3 summarizes the mechanisms and comparison role of each method.

All methods use the observation variables in Eq. (3), the same continuous action limits, reward, terminal-event definitions, control rate, training layouts, locked manifests, and real simulator interaction budget. No method receives a reference path, map, simulator pose, obstacle state, disturbance label, or future observation.

Table 3 Compared methods, available mechanisms, and comparison purpose.

Method	Available mechanisms						Primary comparison purpose
	Mem.	WM	Ens.	Post. cov.	Unc. pen.	Sel. plan.	
TD3	No	No	No	No	No	No	Feedforward model-free continuous control.
Recurrent PPO	Yes	No	No	No	No	No	Tests whether temporal model-free memory is sufficient under partial observability.
DreamerV3	Yes	Yes	No	No	No	No	Tests latent imagination without the proposed uncertainty mechanisms.
DreamerV3-CU	Yes	Yes	Yes	No	Yes	No	Tests conservative uncertainty-aware imagination without posterior-confidence input or deployment replanning.
UG-BWM	Yes	Yes	Yes	Yes	Yes	Yes	Tests the complete posterior-confidence, predictive-uncertainty, and selective replanning framework.

Mem.: memory; WM: world model; Ens.: ensemble; Post. cov.: posterior covariance supplied to the policy; Unc. pen.: uncertainty penalty; Sel. plan.: selective planner.

The visual encoder and auxiliary-input encoders use comparable capacity where this does not alter the defining algorithm. TD3 receives the current encoded observation and contains no recurrent state. Recurrent PPO uses a GRU after the shared observation encoder. DreamerV3 retains its recurrent latent state and standard imagined actor–critic objective. DreamerV3-CU uses the same ensemble size, bootstrap procedure, predictive uncertainty calculation, conservative collision estimate, and imagined penalty as UG-BWM, but does not expose posterior covariance to its actor or critic and does not use deployment-time candidate planning.

Every method receives the same number of real environment transitions. World-model methods may perform latent imagination and additional gradient updates without consuming simulator interactions; their additional computation is reflected in the reported training time rather than treated as environment experience. Hyperparameters are selected using the same model-selection manifest and a comparable number of tuning trials. Parameter count, training time, and inference latency are reported with the navigation results.

The primary confirmatory system comparison is UG-BWM versus DreamerV3-CU because it is the closest controlled baseline. Planned secondary comparisons evaluate UG-BWM against DreamerV3, recurrent PPO, and TD3.

5.5 Controlled Mechanism Tests

Four controlled tests isolate the UG-BWM mechanisms that cannot be separated through external baseline comparisons.

5.5.1 Posterior-confidence input.

Identical frozen UG-BWM checkpoints and actor weights are evaluated with the correct posterior log-covariance, a constant calibration-median covariance, covariance vectors shuffled within each condition and route class, and a separately trained architecture without covariance input. The first three variants preserve h_t, μ_t^q, actor weights, and paired episode manifests, thereby isolating the state-dependent confidence signal. Navigation outcomes, near-term failure ranking, and responses to degradation onset are reported in Fig. 6 and Table 8.

5.5.2 Predictive-uncertainty composition.

The proposed combined uncertainty score is compared with a single predictive head, ensemble disagreement alone, and conditional predictive variance alone. The single-head architecture is trained separately, whereas the ensemble variants share predictive checkpoints and use matched actor–critic initializations, imagined starting states, update budgets, and deployment settings. A complementary 2×2 diagnostic crosses familiar and unseen structure with nominal and degraded vision to examine the empirical response of each uncertainty component. Results are presented in Fig. 8 and Table 8.

5.5.3 Predictive-uncertainty objective.

The proposed state-dependent penalty is compared with no uncertainty penalty, a magnitude-matched constant penalty, shuffled uncertainty values, and a variant without conservative collision adjustment. All variants use the same frozen world-model checkpoint, posterior starting states, matched actor–critic initialization, imagined batches, and update budget; deployment replanning is disabled. The constant and shuffled controls distinguish generic conservatism from transition-specific uncertainty guidance, while the final variant isolates conservative collision handling. Results are reported in Fig. 9 and Table 8.

5.5.4 Replanning trigger.

Identical frozen UG-BWM checkpoints and a fixed candidate planner are evaluated using no replanning, random activation, periodic activation, collision-risk activation, predictive-uncertainty activation with and without hysteresis, and planning at every decision step. All active strategies share the candidate budget, planning horizon, sampling covariance, scoring function, and receding-horizon execution. Competing activation parameters are selected on the model-selection manifest to approximate the uncertainty-trigger activation rate and are then frozen for testing. Navigation performance, planning frequency, trigger switching, and latency are reported in Fig. 10 and Table 9.

5.6 Evaluation Metrics

Each episode is assigned one mutually exclusive terminal outcome: success, collision, timeout, permitted-flight-boundary violation, or abnormal termination. The first terminal event determines the outcome, with collision and boundary violation taking precedence over simultaneous goal completion. Success rate is the primary navigation endpoint, while collision rate and SPL measure safety and path efficiency.

Posterior confidence is summarized by the mean diagonal variance of Σ_t^q. Its decision relevance is assessed through near-term failure rates across uncertainty deciles, its response around visual-degradation and camera-blackout onset, and the controlled covariance interventions in Section 5.5. Near-term failure denotes a collision or boundary violation within the following second. These diagnostics evaluate ranking and state dependence rather than full probabilistic calibration of the posterior covariance.

Multi-step predictive accuracy is evaluated at $h \in \{1, 5, 10, 15, 20, 30\}$ using fixed recorded action sequences. Each rollout begins from the same posterior state and receives no future observations; subsequent observations are used only to construct evaluation targets. The reported metrics are validation-scale-normalized latent prediction error, reward MAE, and collision Brier score. The Brier score uses the ensemble-mean collision probability rather than the conservative collision estimate.

Predictive-uncertainty quality is evaluated using the association between rollout uncertainty $R_t^{(H_u)} = \max_{0 \le k < H_u} U_{t+k}$ and realized rollout error $E_t^{(H_u)} = H_u^{-1} \sum_{k=1}^{H_u} E_{z,t}(k)$. Spearman correlation measures uncertainty–error ranking, while risk–coverage curves and AURC assess selective prediction quality. Prediction-interval calibration error is evaluated at nominal coverage levels $\{0.50, 0.70, 0.90, 0.95\}$ using the propagated predictive mean and covariance. This evaluates the first-two-moment Gaussian approximation rather than the complete ensemble mixture. Failure AUROC measures whether predictive uncertainty anticipates collisions or boundary violations.

Planning outcomes include activation frequency, realized rollout error at activated states, and trigger switch rate. Computational performance is reported through decision-module and end-to-end median and P95 latency, deadline-miss rate, parameter count, and training duration. Runtime conclusions are restricted to the stated workstation configuration and 10-Hz high-level control loop.

5.7 Statistical Analysis

The complete episode is the primary unit for navigation outcomes; successive control steps are not treated as independent observations. Binary outcomes are analysed using mixed-effects logistic regression. The system-level model includes method, evaluation condition, and route class as fixed effects, with random intercepts for trained checkpoint and paired manifest entry. The primary confirmatory contrast compares UG-BWM with DreamerV3-CU, followed by planned comparisons with DreamerV3, recurrent PPO, and TD3.

Mechanism-specific models include covariance intervention, predictive-uncertainty variant, imagined objective variant, or replanning strategy, together with the relevant evaluation condition or route class and their interaction. The separately trained no-covariance architecture is reported as a complementary comparison because it

changes both the available input and the learned parameters. The 2×2 uncertainty-component diagnostic estimates the effects of visual condition, structural condition, and their interaction separately for normalized ensemble disagreement and conditional predictive variance. Horizon-wise quantities are summarized within complete trajectories.

SPL, predictive errors, uncertainty metrics, planning frequency, and runtime are reported with hierarchical bootstrap intervals based on 10,000 replicates, resampling checkpoints before complete episodes or trajectories. Holm correction is applied separately to the system-level, posterior-confidence, predictive-uncertainty, imagined objective, and replanning-trigger comparison families. Confirmatory tests are two-sided at $\alpha = 0.05$. Binary effects are reported as absolute rate differences and adjusted odds ratios with 95% confidence intervals. Continuous effects are reported as mean differences with 95% confidence intervals, except for latency and other time-based outcomes, which are reported as median differences with 95% confidence intervals. Interpretation considers effect magnitude, interval width, seed consistency, and agreement across paired conditions rather than statistical significance alone.

5.8 Implementation, Calibration, and Reproducibility

Table 4 reports the software, model, training, calibration, planning, and evaluation settings. Values selected through model selection remain fixed during locked testing.

Input-normalization statistics are estimated from the training partition and frozen before validation. For each uncertainty-aware checkpoint, Q_{epi} and Q_{ale} are estimated through a no-gradient pass over the fixed calibration manifest. The raw uncertainty components are pooled across episodes, rollout starts, and prediction steps; the references are recomputed once after checkpoint selection and then frozen for locked testing.

Checkpoint selection maximizes mean SPL on the model-selection manifest, with collision rate as the tie breaker. Penalty coefficients, planning settings, and trigger thresholds are selected on the same manifest and remain fixed during locked testing. Evaluation uses deterministic policy actions without dropout, gradient updates, or online adaptation, while the predictive ensemble remains active for uncertainty estimation. Reported end-to-end latency includes preprocessing, posterior inference, rollout monitoring, triggered planning, action transmission, and PX4 communication.

6 Results

Overall metrics aggregate the six locked-test conditions (C1–C6) across five independent training seeds, with 60 episodes per condition and seed. Condition-specific results are identified in the corresponding tables and figures.

6.1 System-Level Navigation Performance

Table 5 reports success under the six locked-test conditions together with overall collision rate, SPL, and long-route success. UG-BWM is compared with TD3, recurrent PPO, DreamerV3, and DreamerV3-CU using identical paired manifests.

Table 4 Implementation, calibration, and evaluation settings.

Group	Setting	Value
Software	Simulator, flight stack, middleware, operating system, and precision	Gazebo Harmonic; PX4 SITL; ROS 2 Jazzy; Micro XRCE-DDS; Ubuntu 24.04; FP32.
Hardware	CPU, GPU, host memory, and GPU memory	Intel Core i9-13900K; NVIDIA RTX 4090 (24GB); 64GB RAM.
Sensing and control	RGB resolution; camera, estimator, and high-level control rates	64×64 pixels; 15 Hz; 50 Hz; 10 Hz.
Action limits	Body-frame velocity and yaw rate	$v_{\max} = 2.0$ m/s; $\dot{\psi}_{\max} = 1.2$ rad/s.
Encoders	Visual, auxiliary, action, and missing-camera representations	Impala CNN with three residual blocks; MLP auxiliary encoders; learned 64-dimensional no-camera embedding.
Latent model	Deterministic dimension, stochastic dimension, and log-standard-deviation bounds	$d_h = 512$; $d = 128$; $\log \sigma \in [-8, 2]$.
Ensemble	Predictive members and sequence-bootstrap probability	$M = 5$; $p_{\text{boot}} = 0.70$.
Replay and training	Replay capacity, sequence length, batch size, and interaction budget	5×10^5 transitions; 50 steps; 32 sequences; 1.5×10^6 real simulator steps.
Optimization	Learning rates, optimizer, gradient clipping, and target update	World model 3×10^{-4}; actor and critic 1×10^{-4}; Adam; clipping at 100; target rate 0.01.
Imagination	Horizon and discount	$H = 15$; $\gamma = 0.99$.
World-model losses	$\lambda_I, \lambda_\eta, \lambda_g, \lambda_{\text{KL}}, \lambda_r, \lambda_c$	1.0, 0.5, 0.5, 1.0, 1.0, 1.0.
Decision objective	$\lambda_{\text{risk}}, \lambda_{\text{unc}}, \alpha, \beta_c$	0.3, 0.15, 0.5, 0.8.
Calibration	Manifest size, rollout starts, horizon, quantile, and refresh interval	90 episodes; 1800 starts; $H = 15$; empirical 0.95 quantile; every 500 world-model updates.
Replanning	$H_{\text{mon}}, H_p, N, \Sigma_{\text{plan}}$	8; 12; 64; $0.2^2 I$.
Trigger thresholds	$\tau_{\text{on}}, \tau_{\text{off}}$ and collision-trigger thresholds	$\tau_{\text{on}} = 0.70$, $\tau_{\text{off}} = 0.45$; collision threshold selected on model-selection manifest.
Visual severity	Fog, illumination, motion blur, and occlusion ranges (clear $\rightarrow$ severe)	Fog visibility (m): [200, 100, 50, 10]; Illumination multiplier: [1.0, 0.6, 0.3, 0.1]; Blur kernel size: [1, 7, 11, 15]; Occlusion fraction: [0.0, 0.1, 0.2, 0.3].
Camera blackout	Blackout duration and recovery settings	Duration: 5–20 s with 10 s median; Recovery: gradual over 2–5 s; Scheduled onset: random within first half of episode.
Evaluation	Training seeds, episodes per primary condition, and prediction horizons	5; 60 per seed; $h \in \{1, 5, 10, 15, 20, 30\}$.

Figure 3 presents the condition-wise results with confidence intervals and shows the variation across independently trained seeds. The estimated overall success difference

Table 5 Navigation performance across the locked-test conditions. Success rates are reported for C1–C6; the remaining columns aggregate all primary-test episodes.

Method	C1	C2	C3	C4	C5	C6	Collision	SPL	Long-route success
TD3	0.52	0.31	0.28	0.24	0.22	0.18	0.39	0.34	0.17
Recurrent PPO	0.63	0.44	0.41	0.36	0.33	0.30	0.30	0.42	0.27
DreamerV3	0.70	0.52	0.48	0.43	0.40	0.36	0.24	0.49	0.33
DreamerV3-CU	0.75	0.59	0.54	0.48	0.45	0.43	0.20	0.54	0.38
UG-BWM	0.81	0.68	0.63	0.58	0.55	0.52	0.15	0.62	0.47

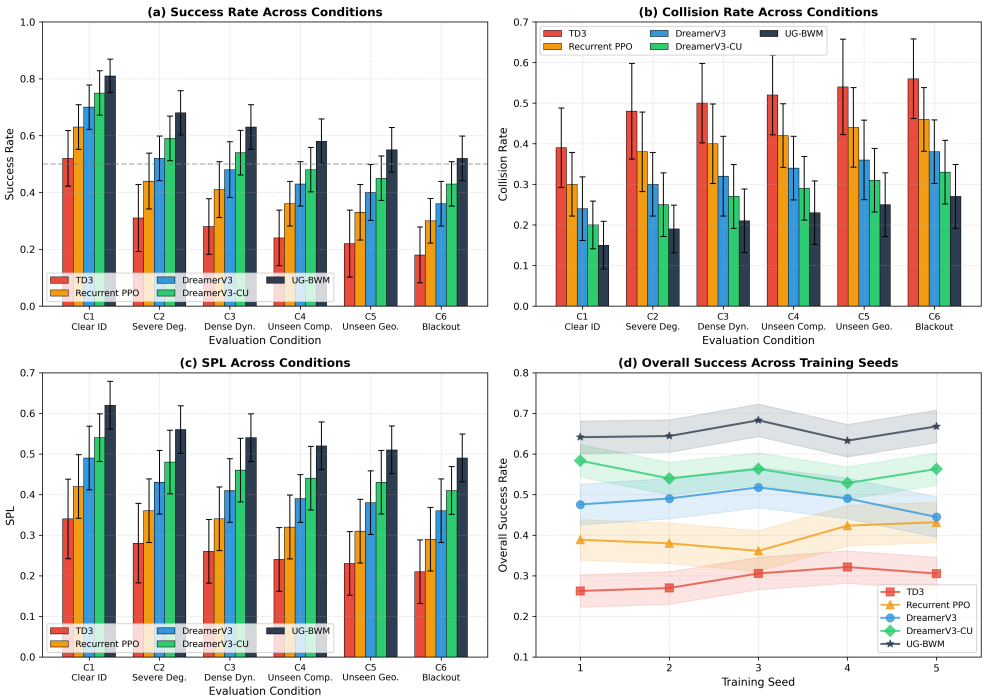


Fig. 3 System-level navigation performance. Panels (a)-(c) compare success, collision rate, and SPL across the six locked-test conditions; error bars indicate 95% confidence intervals. Panel (d) shows seed-level variation in overall success with uncertainty bands.

between UG-BWM and DreamerV3-CU was 0.09, while the corresponding collision-rate and SPL differences were -0.05 and 0.08, respectively. Compared with TD3, UG-BWM achieved a 34 percentage-point improvement in success rate (0.63 vs. 0.29).

The planned contrasts are reported in Table 6. The primary comparison uses DreamerV3-CU because it contains uncertainty-aware imagined learning but excludes explicit posterior-confidence input and deployment-time replanning. UG-BWM improved success, collision rate, and SPL relative to all four baselines. The largest success difference was observed against TD3, whereas DreamerV3-CU provided the closest comparison.

Table 6 Planned system-level contrasts. Differences are calculated as UG-BWM minus the stated baseline and reported on the 0–1 rate scale.

Contrast	Success Δ	Adjusted OR	95% CI	Collision Δ	SPL Δ	Corrected p
UG-BWM vs. TD3	0.34	4.12	[3.21, 5.28]	-0.24	0.28	< 0.001
UG-BWM vs. recurrent PPO	0.22	2.78	[2.22, 3.48]	-0.15	0.20	< 0.001
UG-BWM vs. DreamerV3	0.15	2.05	[1.67, 2.52]	-0.09	0.13	< 0.001
UG-BWM vs. DreamerV3-CU	0.09	1.48	[1.21, 1.81]	-0.05	0.08	0.001

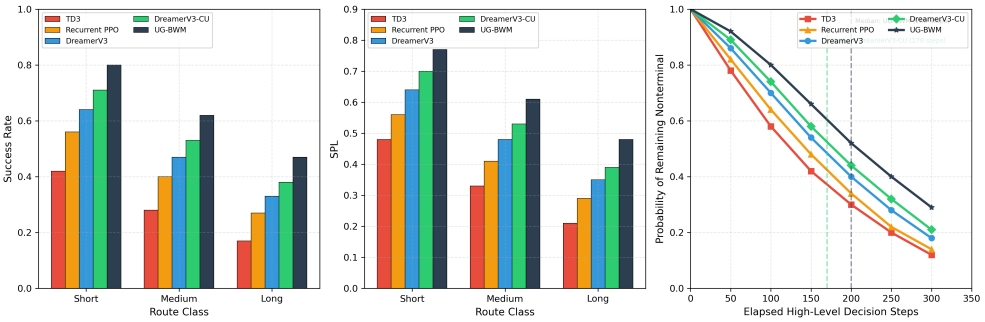


Fig. 4 Effect of mission horizon on navigation performance. Panels (a) and (b) report success rate and SPL for short, medium, and long routes. Panel (c) shows the empirical probability of remaining nonterminal as the number of high-level decision steps increases; dashed vertical lines indicate the median time to the first terminal event for UG-BWM and DreamerV3-CU.

6.2 Mission-Horizon Performance

Figure 4 examines navigation performance across short, medium, and long routes. Success decreased with route length for every method. UG-BWM achieved success rates of 0.80, 0.62, and 0.47 on short, medium, and long routes, respectively, compared with 0.70, 0.53, and 0.38 for DreamerV3-CU. The corresponding UG-BWM advantages were 0.10, 0.09, and 0.09, showing that the improvement was retained as mission length increased.

SPL also decreased with route length, while UG-BWM remained higher than DreamerV3-CU in every route class. UG-BWM obtained values of 0.77, 0.61, and 0.48 on short, medium, and long routes, respectively, compared with 0.70, 0.53, and 0.39 for DreamerV3-CU. The corresponding averages match the overall SPL values of 0.62 and 0.54 reported in Table 5.

Figure 4(c) shows the empirical probability of remaining nonterminal as the number of high-level decision steps increases. UG-BWM retained the highest nonterminal probability throughout the evaluated duration. The median time to the first terminal event was approximately 200 high-level steps for UG-BWM and 170 steps for DreamerV3-CU.

6.3 Robustness to Visual Degradation

Success declined as fog, illumination reduction, motion blur, and partial occlusion became more severe. UG-BWM maintained the highest success rate at every tested

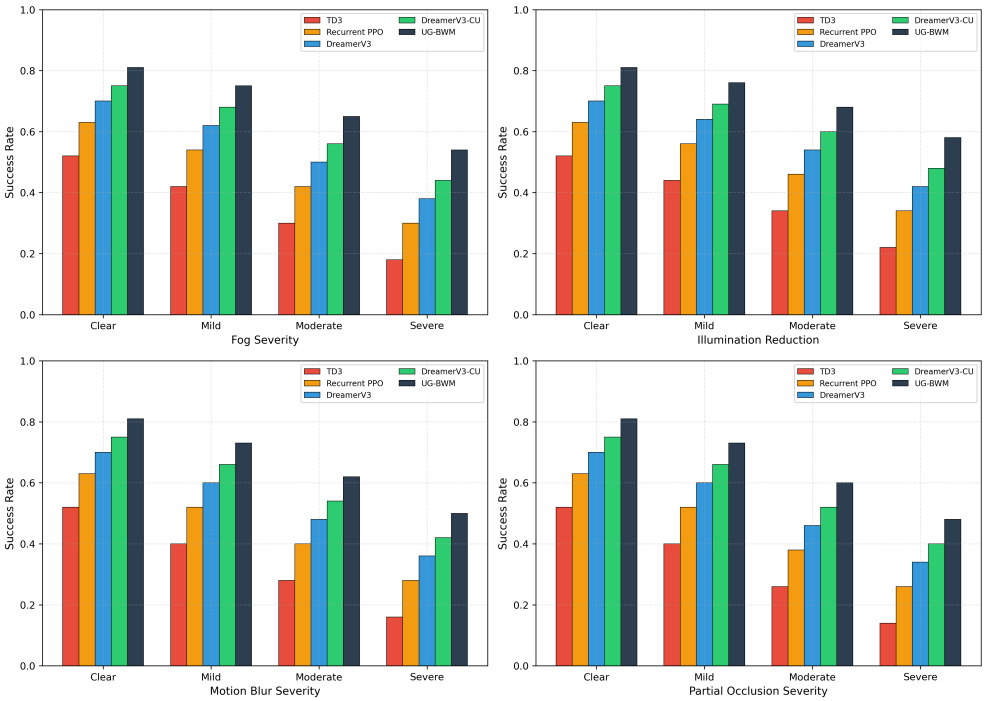


Fig. 5 Success rate under increasing fog, illumination reduction, motion blur, and partial occlusion severity.

severity level, as shown in Fig. 5. Under the combined Test-OOD stress condition, which includes unseen geometry and compound visual degradation, UG-BWM achieved a success rate of 0.49, a collision rate of 0.29, and an SPL of 0.38.

6.4 Posterior-Confidence Evaluation

Near-term failure probability increased from 0.05 in the lowest posterior-uncertainty decile to 0.55 in the highest decile. Posterior uncertainty also increased to 0.72 during severe visual degradation and 0.85 during camera blackout. In the controlled interventions, the correct-covariance variant improved success by 0.06 over the constant-covariance control, 0.04 over the shuffled-covariance control, and 0.09 over the separately trained no-covariance model. These results are presented in Fig. 6 and Group A of Table 8.

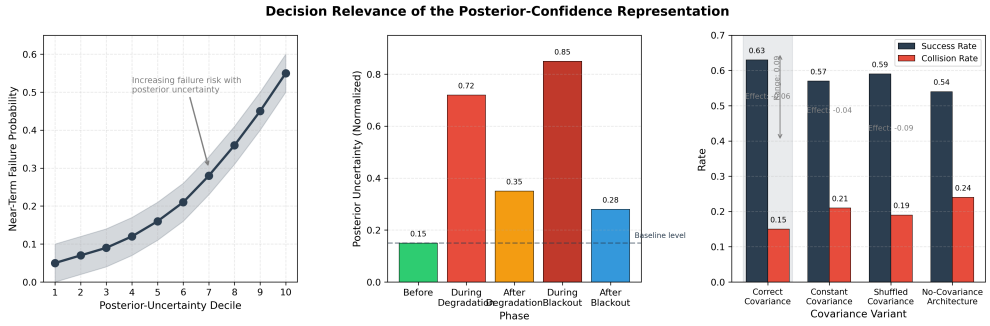


Fig. 6 Posterior uncertainty across failure-risk deciles, visual-evidence changes, and controlled covariance interventions.

6.5 Predictive-Uncertainty Evaluation

6.5.1 Reliability across imagined horizons.

Prediction errors increased with rollout depth for both methods, while uncertainty–error correlation and failure AUROC decreased. UG-BWM maintained lower prediction errors and stronger uncertainty-quality metrics at every evaluated horizon, including beyond the actor-training horizon $H = 15$.

At $h = 15$, UG-BWM achieved a latent prediction error of 0.52, an uncertainty–error correlation of 0.55, a prediction-interval coverage error of 0.07, and an AURC of 0.28. The corresponding DreamerV3-CU values were 0.58, 0.47, 0.11, and 0.36. At $h = 30$, UG-BWM retained lower latent prediction error, coverage error, and AURC, with values of 0.89, 0.12, and 0.42 compared with 0.92, 0.18, and 0.52. Its uncertainty–error correlation also remained higher at 0.41 compared with 0.34.

Using the rollout uncertainty statistic $R_t^{(H_u)}$ with $H_u = 15$, UG-BWM achieved near-term failure AUROC values of 0.86 under clear ID conditions, 0.78 under severe visual degradation, 0.74 under unseen geometry, and 0.72 during camera blackout. The complete horizon-wise results are reported in Table 7.

6.5.2 Response to visual and structural shift.

Ensemble disagreement responded more strongly to structural novelty, increasing from 0.15 in ID-clear states to 0.52 in OOD-clear states and 0.78 in OOD-degraded states. The estimated visual and structural effects were 0.28 and 0.35, with an interaction of -0.04. Conditional predictive variance increased from 0.12 in ID-clear states to 0.38 in ID-degraded states and 0.55 in OOD-degraded states. Its visual and structural effects were 0.23 and 0.20, with an interaction of -0.06.

The combined predictive-uncertainty score increased from 0.14 in ID-clear states to 0.43, 0.46, and 0.70 in ID-degraded, OOD-clear, and OOD-degraded states, respectively. Realized rollout error followed the same condition ordering, increasing from 0.22 to 0.45, 0.55, and 0.78. Both quantities were highest when visual degradation and structural novelty occurred simultaneously.

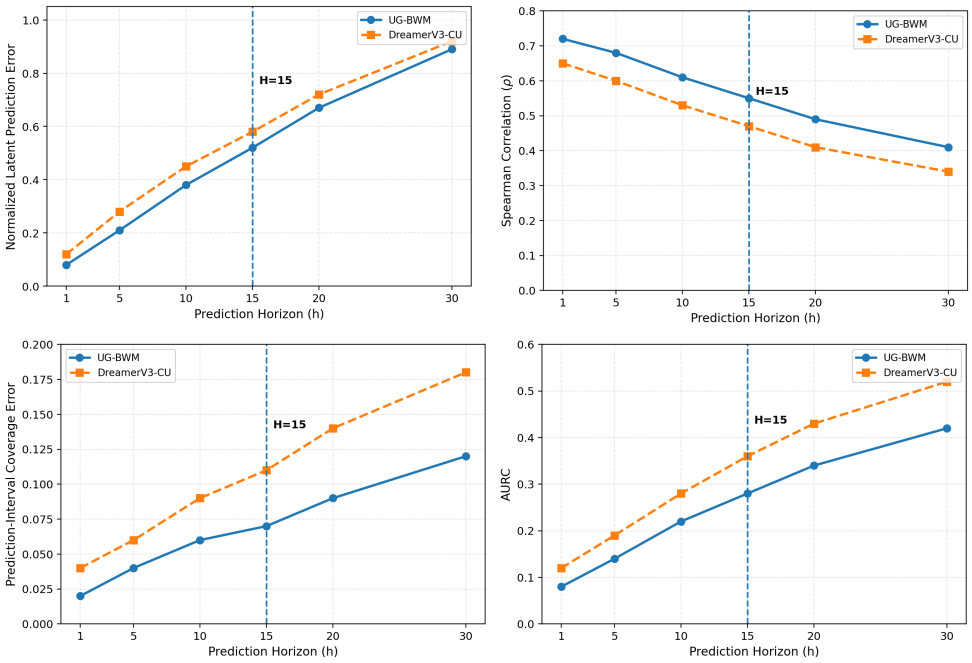


Fig. 7 Predictive accuracy and uncertainty quality across imagined horizons. The vertical marker at $H = 15$ indicates the actor-training horizon.

Table 7 Predictive accuracy and uncertainty quality across imagined horizons for UG-BWM and DreamerV3-CU.

Method	Metric	$h = 1$	$h = 5$	$h = 10$	$h = 15$	$h = 20$	$h = 30$
UG-BWM	Latent prediction error	0.08	0.21	0.38	0.52	0.67	0.89
	Reward MAE	0.06	0.14	0.23	0.31	0.38	0.49
	Collision Brier score	0.04	0.08	0.13	0.18	0.22	0.29
	Uncertainty–error Spearman ρ	0.72	0.68	0.61	0.55	0.49	0.41
	Prediction-interval coverage error	0.02	0.04	0.06	0.07	0.09	0.12
	AURC	0.08	0.14	0.22	0.28	0.34	0.42
	Failure AUROC	0.86	0.81	0.76	0.71	0.66	0.58
DreamerV3-CU	Latent prediction error	0.12	0.28	0.45	0.58	0.72	0.92
	Reward MAE	0.08	0.18	0.28	0.36	0.44	0.54
	Collision Brier score	0.06	0.11	0.16	0.22	0.27	0.34
	Uncertainty–error Spearman ρ	0.65	0.60	0.53	0.47	0.41	0.34
	Prediction-interval coverage error	0.04	0.06	0.09	0.11	0.14	0.18
	AURC	0.12	0.19	0.28	0.36	0.43	0.52
	Failure AUROC	0.79	0.74	0.68	0.63	0.58	0.50

6.6 Uncertainty-Guided Imagined Learning

With deployment replanning disabled, the state-dependent penalty achieved a success rate of 0.60, a collision rate of 0.18, an SPL of 0.59, and a long-route success rate of 0.43. Success exceeded the no-penalty, matched-constant, and shuffled-uncertainty controls by 0.08, 0.06, and 0.05, respectively. Removing the conservative collision adjustment

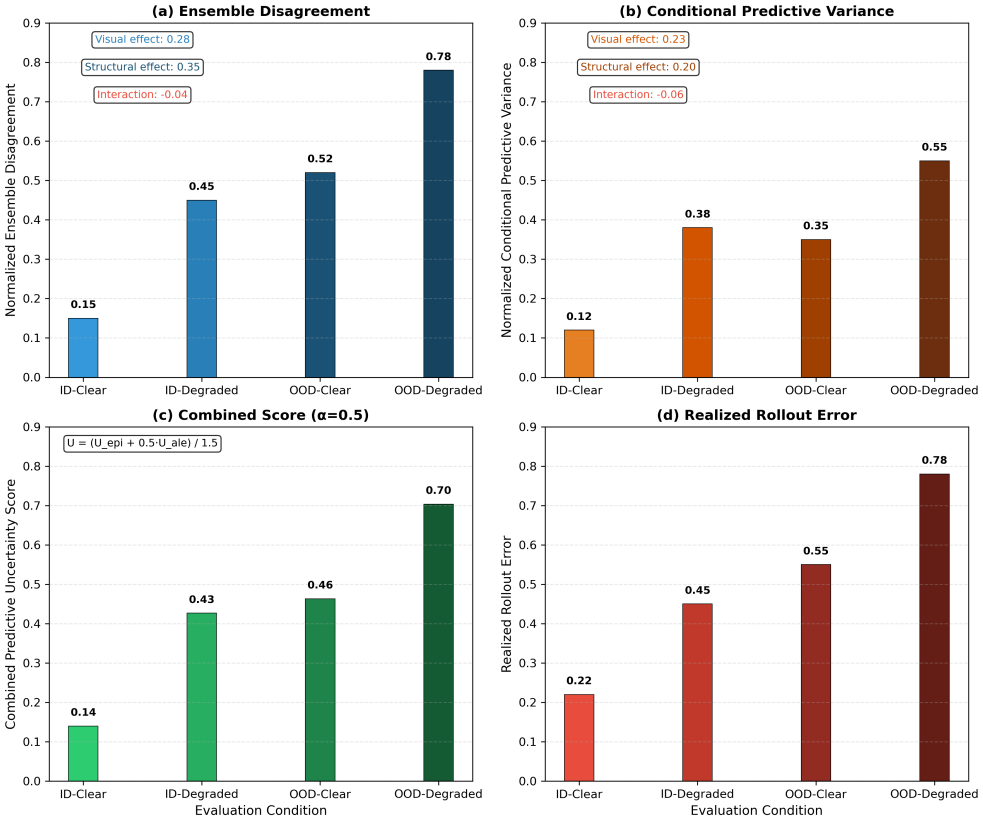


Fig. 8 Normalized ensemble disagreement, conditional predictive variance, combined predictive uncertainty, and realized rollout error under visual degradation and structural novelty.

reduced success to 0.58 and increased the collision rate to 0.21. The advantage of the state-dependent penalty remained across short, medium, and long routes. Figure 9 and Group C of Table 8 report the complete results. The state-dependent variant corresponds to the no-replanning configuration in Table 9.

6.7 Selective Replanning and Computational Cost

The predictive-uncertainty trigger activated the candidate planner on 18% of control steps and achieved a success rate of 0.63, a collision rate of 0.15, and an SPL of 0.62. Random, periodic, and collision-risk activation used comparable planning rates of 0.19, 0.18, and 0.17, but achieved lower success rates of 0.61, 0.61, and 0.62, respectively.

The predictive-uncertainty trigger selected states with a mean realized rollout error of 0.89, compared with 0.76 for random activation, 0.74 for periodic activation, and 0.80 for collision-risk activation. Removing hysteresis increased the switch rate from 0.08 to 0.23 and the planning rate from 0.18 to 0.24.

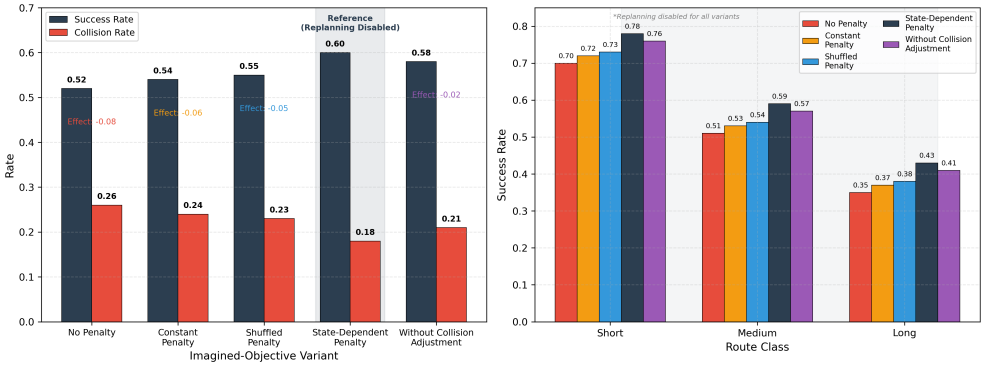


Fig. 9 Navigation effects of the imagined objective controls with deployment replanning disabled. Panel (a) reports aggregate success and collision rates, and panel (b) reports success across route classes.

Table 8 Unified ablation results. Groups A and B use complete UG-BWM as the reference; Group C disables deployment replanning. Metrics aggregate C1–C6.

A. Posterior-confidence interface

Variant	Success	Collision	SPL	Long-route success	Effect vs. reference	Corrected p
Correct covariance	0.63	0.15	0.62	0.47	Reference	–
Constant covariance	0.57	0.21	0.56	0.40	-0.06	< 0.001
Shuffled covariance	0.59	0.19	0.58	0.42	-0.04	< 0.001
No-covariance architecture	0.54	0.24	0.53	0.37	-0.09	< 0.001

B. Predictive-uncertainty composition

Variant	Success	Collision	Uncertainty–error ρ	AURC	Effect vs. reference	Corrected p
Single predictive head	0.57	0.20	0.38	0.39	-0.06	< 0.001
Ensemble disagreement only	0.59	0.18	0.49	0.32	-0.04	< 0.001
Conditional variance only	0.60	0.17	0.51	0.30	-0.03	0.002
Combined score	0.63	0.15	0.55	0.28	Reference	–

C. Imagined-objective controls (Replanning disabled)

Variant	Success	Collision	SPL	Long-route success	Effect vs. reference	Corrected p
No uncertainty penalty	0.52	0.26	0.51	0.35	-0.08	< 0.001
Matched constant penalty	0.54	0.24	0.53	0.37	-0.06	< 0.001
Shuffled uncertainty penalty	0.55	0.23	0.54	0.39	-0.05	< 0.001
State-dependent penalty	0.60	0.18	0.59	0.43	Reference	–
Without conservative collision adjustment	0.58	0.21	0.57	0.41	-0.02	0.031

Always-plan matched the success, collision rate, and SPL of the proposed trigger but activated the planner at every control step. The proposed trigger therefore achieved the same navigation performance with 82% fewer planner activations and reduced decision-module P95 latency from 71 ms to 48 ms.

Table 9 reports the complete replanning and computational results. UG-BWM required 14.6M parameters and 8.4 hours of training. Its complete observation-to-command latency was 36 ms at the median and 63 ms at P95, with a 1.1% deadline-miss rate under the 10-Hz control period. These measurements apply to the workstation configuration in Table 4.

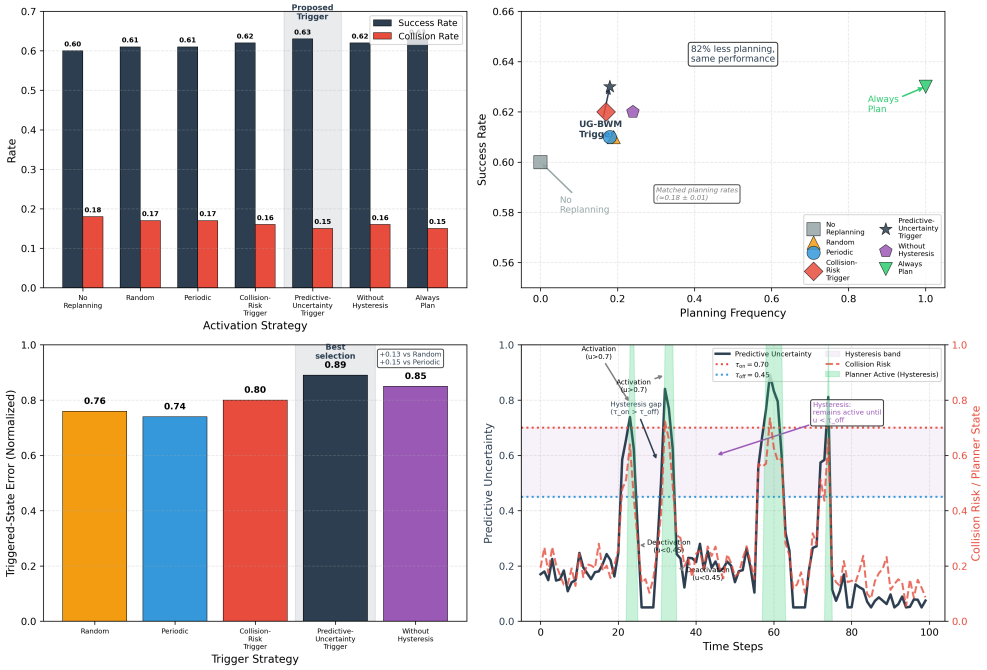


Fig. 10 Navigation performance, planning frequency, trigger quality, and hysteretic activation under the replanning strategies.

Table 9 Selective-replanning outcomes and computational cost. Part A compares activation strategies across C1–C6; Part B reports method-level training and inference cost.

A. Replanning activation strategies

Strategy	Success	Collision	SPL	Planning rate	Triggered-state error	Switch rate	Decision-module P95 latency (ms)
No replanning	0.60	0.18	0.59	0.00	–	–	42
Random	0.61	0.17	0.60	0.19	0.76	0.38	48
Periodic	0.61	0.17	0.60	0.18	0.74	0.35	47
Collision-risk trigger	0.62	0.16	0.61	0.17	0.80	0.28	46
Predictive-uncertainty trigger	0.63	0.15	0.62	0.18	0.89	0.08	48
Without hysteresis	0.62	0.16	0.61	0.24	0.85	0.23	53
Always plan	0.63	0.15	0.62	1.00	0.86	0.00	71

B. Method-level computational cost

Method	Parameters	Training time	End-to-end median latency (ms)	End-to-end P95 latency (ms)	Deadline misses
TD3	4.2M	2.1h	18	32	0.1%
Recurrent PPO	5.8M	2.8h	22	38	0.2%
DreamerV3	12.4M	6.5h	28	52	0.5%
DreamerV3-CU	13.1M	7.2h	32	58	0.8%
UG-BWM	14.6M	8.4h	36	63	1.1%

7 Discussion

7.1 Navigation under Partial Observability and Visual Degradation

UG-BWM maintained its advantage across the locked-test conditions and route classes. The largest differences from DreamerV3-CU appeared under visual degradation, missing observations, dynamic obstacles, and unseen geometry. Explicit uncertainty information is therefore most useful when the current state and its future evolution cannot be inferred reliably from observation history alone.

Long-horizon navigation is achieved through repeated finite-horizon imagination and posterior correction, not through open-loop prediction over the complete mission. Performance decreased with route length for all methods, but the advantage of UG-BWM remained stable. The method also retained the highest success across the visual-degradation sweeps, although its performance declined under severe and combined shifts. UG-BWM therefore reduces sensitivity to degraded sensing but does not remove it.

7.2 Decision Relevance of Posterior Confidence

Posterior covariance changed with near-term failure risk and the quality of the available visual evidence. The controlled interventions provide stronger evidence of its decision relevance. Replacing the correct covariance with constant or shuffled values reduced navigation performance. Because shuffling preserved the covariance distribution while breaking its association with the current belief, the result indicates that state-dependent confidence matters more than simply providing an additional input vector.

The no-covariance architecture also performed worse, supporting the explicit confidence interface. These findings do not establish that posterior covariance is a calibrated probability of navigation failure. It is better viewed as a decision-relevant confidence signal whose reliability may weaken under distribution shift [59, 60].

7.3 Predictive-Uncertainty Reliability and Composition

Prediction quality deteriorated as the rollout horizon increased. Nevertheless, predictive uncertainty remained positively associated with realized error beyond the actor-training horizon. This association weakened at longer horizons, so the uncertainty score can rank less reliable rollouts but should not be treated as equally informative at all rollout depths.

The two uncertainty components responded differently to distribution shift. Ensemble disagreement showed the stronger response to structural novelty, whereas conditional predictive variance responded more strongly to degraded vision. Both reached their highest values when visual and structural shifts occurred together, where rollout error was also largest.

The combined score performed better than either component alone, supporting their joint use. The distinction between disagreement and conditional variance remains

operational, however. The results do not establish a unique physical decomposition into epistemic and aleatoric uncertainty.

7.4 Uncertainty-Guided Imagined Learning

The matched-constant and shuffled controls distinguish transition-specific uncertainty guidance from general conservatism. Both performed worse than the state-dependent penalty. The constant control preserved a similar average penalty, while the shuffled control preserved its distribution but removed its association with individual imagined transitions. The benefit therefore depends on where uncertainty occurs within the rollout, not only on reducing imagined rewards overall.

Removing the conservative collision adjustment also reduced performance. Predictive uncertainty and collision probability provide different information: predictive uncertainty reflects reduced confidence in the latent transition, whereas the collision head estimates a specific adverse event. Their separate penalties allow the policy to account for both model reliability and predicted safety risk, consistent with uncertainty-aware MBRL [21].

The benefit of the state-dependent penalty remained across all route classes, indicating that it supports policy learning generally rather than only during long missions.

7.5 Selective Replanning and Computational Trade-Off

At comparable planning rates, the predictive-uncertainty trigger selected states with higher realized rollout error than random, periodic, or collision-risk activation. It also achieved the strongest navigation performance among the matched-rate strategies. Predictive uncertainty therefore provides useful information about when the nominal rollout is unreliable and additional planning is beneficial.

The proposed trigger matched always-plan performance while activating the planner on only 18% of control steps. This result is specific to the tested candidate budget, thresholds, environments, and hardware; it does not imply that selective planning will always equal continuous planning. Hysteresis further reduced rapid switching without lowering navigation performance.

UG-BWM required more parameters and training time than the baselines. Its measured latency supported the evaluated 10-Hz control loop on the RTX 4090 workstation, but these results do not establish real-time feasibility on embedded UAV hardware. The latency advantage over always-plan arises from less frequent planner use rather than a smaller world model.

7.6 Limitations and Future Evaluation

The study is limited to simulation. Gazebo and PX4 SITL provide controlled and repeatable testing, but they cannot reproduce all camera artifacts, aerodynamic disturbances, estimator failures, and timing constraints encountered during physical flight. Hardware experiments are needed to evaluate UG-BWM under real sensor and dynamics shifts.

The predictive ensemble is represented by a single Gaussian through first-two-moment matching. This supports efficient rollouts but may hide multiple plausible future outcomes. Multimodal predictive beliefs could preserve these alternatives at a higher computational cost.

The uncertainty-normalization references and replanning thresholds are fixed during testing. Their reliability may decrease under shifts that differ from the calibration data or over longer rollouts [60]. Adaptive calibration should therefore be examined without using privileged test information.

UG-BWM reduces collision risk empirically but does not enforce a hard safety constraint or provide a formal guarantee. Future work should evaluate the method on physical UAVs, examine multimodal beliefs and adaptive calibration, and integrate explicit safety constraints.

8 Conclusion

This work presented UG-BWM for long-horizon UAV navigation under partial observability and visual degradation. The method represents confidence in the current posterior belief, propagates predictive uncertainty through imagined rollouts, uses this uncertainty during policy learning, and selectively activates deployment-time planning.

Across six locked-test conditions, UG-BWM achieved a success rate of 0.63, a collision rate of 0.15, and an SPL of 0.62, outperforming the closest controlled baseline. The controlled tests showed that posterior confidence, combined predictive uncertainty, and the state-dependent uncertainty penalty each contributed to navigation performance. Predictive uncertainty remained associated with rollout error beyond the training horizon, while uncertainty-triggered replanning matched always-plan performance with 82% fewer planner activations.

These results support uncertainty-explicit belief modeling and selective planning for navigation under degraded sensing. Physical UAV experiments and integration with explicit safety constraints are needed to assess the method beyond simulation.

Acknowledgment

The author thanks the Department of Computer Engineering at the University of Basrah for supporting this research. The author also acknowledges the open-source contributions of the Gazebo, PX4, and ROS communities that supported the development of the simulation framework.

Funding

The author declares that no funds, grants, or other support were received during the preparation of this manuscript.

References

[1] Irfan, M., Dalai, S., Trslic, P., Riordan, J., Dooly, G.: LSAF-LSTM-based self-adaptive multi-sensor fusion for robust UAV state estimation in challenging

environments. Machines **13**(2), 130 (2025)

[2] Katkuri, A.V.R., Madan, H., Khatri, N., Abdul-Qawy, A.S.H., Patnaik, K.S.: Autonomous UAV navigation using deep learning-based computer vision frameworks: A systematic literature review. Array **23**, 100361 (2024)

[3] Izakshiri, A., Tajrishi, N.J., Madinei, H.S., Azad, P.A., Ahmadi, S.: A comprehensive survey on autonomous UAV navigation and control systems. International Journal of Dynamics and Control **14**(4), 141 (2026)

[4] Meng, W., Zhang, X., Zhou, L., Guo, H., Hu, X.: Advances in UAV path planning: A comprehensive review of methods, challenges, and future directions. Drones **9**(5), 376 (2025)

[5] Sai, S., Garg, A., Jhawar, K., Chamola, V., Sikdar, B.: A comprehensive survey on artificial intelligence for unmanned aerial vehicles. IEEE Open Journal of Vehicular Technology **4**, 713–738 (2023)

[6] Dutta, S., Gupta, A., Saran, V., Zakhor, A.: Vision-guided outdoor flight and obstacle evasion via reinforcement learning. IEEE Robotics and Automation Letters **11**(2), 1202–1209 (2026)

[7] Hu, Z., Butt, M.Z., Nasir, N., Salleh, M.B.: Vision-based autonomous navigation for quadrotor uavs in unknown and constrained environments. International Journal of Intelligent Robotics and Applications **9**(4), 1743–1754 (2025)

[8] Xiao, J., Zhang, R., Zhang, Y., Feroskhan, M.: Vision-based learning for drones: A survey. IEEE Transactions on Neural Networks and Learning Systems **36**(9), 15601–15621 (2025)

[9] Yu, H., De Wagter, C., de Croon, G.C.H.E.: Mavrl: Learn to fly in cluttered environments with varying speed. IEEE Robotics and Automation Letters **10**(2), 1441–1448 (2025)

[10] Han, H., Cheng, J., Lv, M., Xi, Z.: Autonomous navigation of UAVs in unknown 3d environments using deep reinforcement learning for path planning. IEEE Transactions on Vehicular Technology **PP**(99), 1–13 (2025)

[11] Guo, C., Su, C., Zhang, Z., Han, J., Fang, Y., Liang, X.: PEMTRS: Perception-enhanced memory with temporal region selection for vision-based multirotor navigation. IEEE Robotics and Automation Letters **11**(7), 7884–7891 (2026)

[12] Sheng, Y., Li, J., Liu, H., Xie, X.: Robust UAV navigation under perception uncertainty: A spatiotemporal transformer-enhanced reinforcement learning approach. Advanced Engineering Informatics **73**, 104596 (2026)

[13] Haddad, M.A., Khudher, D.R.: Dual-memory architecture for robust UAV: Navigation integrating lstm and transformer within a PPO framework. International Journal of Robotics and Control Systems **5**(5), 2522–2545 (2025)

[14] Ha, D., Schmidhuber, J.: Recurrent world models facilitate policy evolution. In: Advances in Neural Information Processing Systems, vol. 31, pp. 2450–2462. Curran Associates, Inc., ??? (2018)

[15] Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. In: Proceedings of the 36th International Conference on Machine Learning, vol. 97, pp. 2555–2565. PMLR, ??? (2019)

[16] Hafner, D., Lillicrap, T., Ba, J., Norouzi, M.: Dream to control: Learning behaviors by latent imagination. In: International Conference on Learning Representations (2020). https://openreview.net/forum?id=S1lOTC4tDS

[17] Hansen, N.A., Su, H., Wang, X.: Temporal difference learning for model predictive control. In: Proceedings of the 39th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 162, pp. 8387–8406. PMLR, ??? (2022)

[18] Shanks, S., Embley-Riches, J., Liu, J., Delfaki, A.M., Ciliberto, C., Kanoulas, D.: Dreamernav: Learning-based autonomous navigation in dynamic indoor environments using world models. Frontiers in Robotics and AI **12**, 1655171 (2025)

[19] Bar, A., Zhou, G., Tran, D., Darrell, T., LeCun, Y.: Navigation world models. In: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 15791–15801 (2025)

[20] Zhong, S., Du, X., Xia, K., Jiang, Y., Xue, Y., Gong, S.: Trajectory self-correction and uncertainty estimation for enhanced model-based policy optimization. Expert Systems with Applications **274**, 126993 (2025)

[21] Yu, D., Zou, W., Yang, Y., Ma, H., Li, S.E., Yin, Y., Chen, J., Duan, J.: Safe model-based reinforcement learning with an uncertainty-aware reachability certificate. IEEE Transactions on Automation Science and Engineering **21**(3), 4129–4142 (2023)

[22] Ghambari, S., Golabi, M., Jourdan, L., Lepagnot, J., Idoumghar, L.: UAV path planning techniques: A survey. RAIRO-Operations Research **58**(4), 2951–2989 (2024)

[23] Sheltami, T., Ahmed, G., Ghaleb, M., Mahmoud, A.: UAV path planning and trajectory optimization: A comprehensive survey. Arabian Journal for Science and Engineering **51**(1), 105–145 (2026)

[24] Yin, Y., Wang, Z., Zheng, L., Su, Q., Guo, Y.: Autonomous UAV navigation with adaptive control based on deep reinforcement learning. Electronics **13**(13), 2432 (2024)

[25] Zhang, S., Li, Y., Dong, Q.: Autonomous navigation of UAV in multi-obstacle environments based on a deep reinforcement learning approach. Applied Soft Computing **115**, 108194 (2022)

[26] Jarray, R., Bouallègue, S.: Improved deep q-network-based UAV path planning in partially observable environments. Cluster Computing **28**(16), 1046 (2025)

[27] Boiteau, S., Vanegas, F., Gonzalez, F.: Framework for autonomous UAV navigation and target detection in global-navigation-satellite-system-denied and visually degraded environments. Remote Sensing **16**(3), 471 (2024)

[28] Pan, Z., Zhang, C., Xia, Y., Xiong, H., Shao, X.: An improved artificial potential field method for path planning and formation control of the multi-UAV systems. IEEE Transactions on Circuits and Systems II: Express Briefs **69**(3), 1129–1133 (2021)

[29] Hao, G., Lv, Q., Huang, Z., Zhao, H., Chen, W.: UAV path planning based on improved artificial potential field method. Aerospace **10**(6), 562 (2023)

[30] Liu, J., Yan, Y., Yang, Y., Li, J.: An improved artificial potential field UAV path planning algorithm guided by RRT under environment-aware modeling: Theory and simulation. IEEE Access **12**, 12080–12097 (2024)

[31] Wu, H., Wang, W., Wang, T., Suzuki, S.: Model-free uav navigation in unknown complex environments using vision-based reinforcement learning. Drones **9**(8), 566 (2025)

[32] Imanberdiyev, N., Smith, A., Kovac, M.: Vision-based autonomous UAV navigation in cluttered environments using proximal policy optimization. Journal of Intelligent & Robotic Systems **111**(2), 45 (2025)

[33] Guo, J., Zhou, G., Huang, H., Huang, C.: Advancements in UAV path planning: A deep reinforcement learning approach with soft actor-critic for enhanced navigation. Unmanned Systems **13**(04), 1065–1084 (2025)

[34] Fujimoto, S., Hoof, H., Meger, D.: Addressing function approximation error in actor–critic methods. In: Proceedings of the 35th International Conference on Machine Learning, vol. 80, pp. 1587–1596. PMLR, ??? (2018)

[35] Sharma, G., Jain, S., Sharma, R.S.: Sac-eprb: Soft actor-critic with enhanced prioritized replay buffer for UAV navigation. Intelligent Service Robotics **19**(3), 48 (2026)

[36] AlMahamid, F., Grolinger, K.: Agile dqn: adaptive deep recurrent attention reinforcement learning for autonomous UAV obstacle avoidance. Scientific Reports **15**(1), 18043 (2025)

[37] Liu, S., Zou, S., Chang, X., Liu, H., Zhang, L., Deng, X.: Autonomous navigation of UAV in complex environment: a deep reinforcement learning method based on temporal attention. Applied Intelligence **55**(5), 316 (2025)

[38] Yu, T., Dang, Y., Chen, P., Zhao, D., Liang, R.: A spatial-aware temporal modeling network for imitation learning-based drone navigation. IEEE Transactions on Intelligent Transportation Systems **26**(10), 15518–15528 (2025)

[39] Zhang, H., Liu, F., Zhang, C., Yu, Q.: STAF-Navi: Vision-based spatio-temporal attention fusion navigation framework. IEEE Robotics and Automation Letters **10**(12), 12796–12803 (2025)

[40] Boiteau, S., Vanegas, F., Galvez-Serna, J., Gonzalez, F.: Model-based RL decision-making for UAVs operating in GNSS-denied, degraded visibility conditions with limited sensor capabilities. Drones **9**(6), 410 (2025)

[41] Mao, Y., Zhan, Q., Yang, L., Zhang, C., Xu, G., Shen, R.: The motion estimation of unmanned aerial vehicle axial velocity using blurred images. Drones **8**(7), 306 (2024)

[42] Zhang, J., Lei, C., Dai, C., Hoi, K., Wang, L., Han, Z., Gao, F.: High-speed vision-based flight in clutter with safety-shielded reinforcement learning. IEEE Robotics and Automation Letters **11**(8), 9559–9566 (2026)

[43] Deng, Y., Hu, Y., Yu, F., Zhang, L., Zou, D.: Autonomous FPV flight with translational optical flow and uncertainty mask. IEEE Robotics and Automation Letters **11**(9), 10433–10440 (2026)

[44] Hafner, D., Pasukonis, J., Ba, J., Lillicrap, T.: Mastering diverse control tasks through world models. Nature **640**(8059), 647–653 (2025)

[45] Hafner, D., Lillicrap, T., Norouzi, M., Ba, J.: Mastering Atari with discrete world models. In: International Conference on Learning Representations (2021)

[46] Chua, K., Calandra, R., McAllister, R., Levine, S.: Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In: Advances in Neural Information Processing Systems, vol. 31, pp. 4754–4765 (2018)

[47] Janner, M., Fu, J., Zhang, M., Levine, S.: When to trust your model: Model-based policy optimization. In: Advances in Neural Information Processing Systems, vol. 32, pp. 12498–12513 (2019)

[48] Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive

uncertainty estimation using deep ensembles. In: Advances in Neural Information Processing Systems, vol. 30, pp. 6402–6413 (2017)

[49] Zhao, J., Wang, Y., Cai, Z., Liu, N., Wu, K., Wang, Y.: Learning visual representation for autonomous drone navigation via a contrastive world model. IEEE Transactions on Artificial Intelligence **5**(3), 1263–1276 (2024)

[50] Martinez-Baselga, D., Riazuelo, L., Montano, L.: Rumor: Reinforcement learning for understanding a model of the real world for navigation in dynamic environments. Robotics and Autonomous Systems **191**, 105020 (2025)

[51] Tommaselli, F., Affonso, F., Rocha, A., Capezzuto, G., Sivakumar, A.N., Chowdhary, G., Becker, M.: LeCropFollow: Latent space planning for navigation in unstructured crop fields. IEEE Robotics and Automation Letters **11**(9), 10369–10376 (2026)

[52] Fujii, K., Murata, S.: Real-world robot control by deep active inference with a temporally hierarchical world model. IEEE Robotics and Automation Letters **11**(1), 890–897 (2026)

[53] Yu, T., Thomas, G., Yu, L., Ermon, S., Zou, J.Y., Levine, S., Finn, C., Ma, T.: MOPO: Model-based offline policy optimization. In: Advances in Neural Information Processing Systems, vol. 33, pp. 14129–14142 (2020)

[54] Zhu, J., Du, C., Dullerud, G.E.: Model-based offline reinforcement learning with uncertainty estimation and policy constraint. IEEE Transactions on Artificial Intelligence **5**(12), 6066–6079 (2024)

[55] Yu, T., Kumar, A., Rafailov, R., Rajeswaran, A., Levine, S., Finn, C.: Combo: Conservative offline model-based policy optimization. In: Advances in Neural Information Processing Systems, vol. 34, pp. 28954–28967 (2021)

[56] Zhang, Y., Wang, Y., Yan, P., Wen, W.: Learning safe, optimal, and real-time flight interaction with deep confidence-enhanced reachability guarantee. IEEE Transactions on Intelligent Transportation Systems **26**(12), 23100–23112 (2025)

[57] Zhang, Z., Liu, Q., Li, Y., Lin, K., Li, L.: Safe reinforcement learning in autonomous driving with epistemic uncertainty estimation. IEEE Transactions on Intelligent Transportation Systems **25**(10), 13653–13666 (2024)

[58] Xing, Z., Zheng, Y., Zhang, Q., Ding, Z., Yang, P., Gu, S., Xia, Z., Zhao, D.: Mimir: Hierarchical goal-driven diffusion with uncertainty propagation for end-to-end autonomous driving. IEEE Robotics and Automation Letters **11**(2), 2178–2185 (2026)

[59] Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural

networks. In: Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 1321–1330. PMLR, ??? (2017)

[60] Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., Snoek, J.: Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In: Advances in Neural Information Processing Systems, vol. 32, pp. 13991–14002 (2019)