drones

MDPI

Article

UAV Collision Avoidance in Unknown Scenarios with Causal Representation Disentanglement

Zhun Fan [1,2], Zihao Xia [3,†], Che Lin [3,†], Gaofei Han [3], Wenji Li [2,3], Dongliang Wang [2,3], Yindong Chen [3], Zhifeng Hao [3], Ruichu Cai [4] and Jiafan Zhuang [2,3,*]

1 Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen 518110, China; fanzhun@uestc.edu.cn
2 International Cooperation Base of Evolutionary Intelligence and Robotics, Shantou 515041, China; liwj@stu.edu.cn (W.L.); dlwang@stu.edu.cn (D.W.)
3 College of Engineering, Shantou University, Shantou 515063, China; 22zhxia@stu.edu.cn (Z.X.); 19clin1@stu.edu.cn (C.L.); 22gfhan@stu.edu.cn (G.H.); ydchen@stu.edu.cn (Y.C.); haozhifeng@stu.edu.cn (Z.H.)
4 School of Computer Science, Guangdong University of Technology, Guangzhou 510006, China; cairuichu@gdut.edu.cn
* Correspondence: jfzhuang@stu.edu.cn
† These authors contributed equally to this work.

Abstract: Deep reinforcement learning (DRL) has significantly advanced online path planning for unmanned aerial vehicles (UAVs). Nonetheless, DRL-based methods often encounter reduced performance when dealing with unfamiliar scenarios. This decline is mainly caused by the presence of non-causal and domain-specific elements within visual representations, which negatively impact the learning of policies. Present techniques generally depend on predefined augmentation or regularization methods intended to direct the model toward identifying causal and domain-invariant components, thereby enhancing the model's ability to generalize. However, these manually crafted approaches are intrinsically constrained in their coverage and do not consider the entire spectrum of possible scenarios, resulting in less effective performance in new environments. Unlike prior studies, this work prioritizes representation learning and presents a novel method for causal representation disentanglement. The approach successfully distinguishes between causal and non-causal elements in visual data. By concentrating on causal aspects during the policy learning phase, the impact of non-causal factors is minimized, thereby improving the generalizability of DRL models. Experimental results demonstrate that our technique achieves reliable navigation and effective collision avoidance in unseen scenarios, surpassing state-of-the-art deep reinforcement learning algorithms.

Keywords: deep reinforcement learning; unmanned aerial vehicles; generalization capability; policy learning; causal representation disentanglement

check for updates

Academic Editor: Agostino De Marco

Received: 5 November 2024
Revised: 19 December 2024
Accepted: 20 December 2024
Published: 25 December 2024

Citation: Fan, Z.; Xia, Z.; Lin, C.; Han, G.; Li, W.; Wang, D.; Chen, Y.; Hao, Z.; Cai, R.; Zhuang, J. UAV Collision Avoidance in Unknown Scenarios with Causal Representation Disentanglement. *Drones* **2025**, *9*, 10. https://doi.org/10.3390/drones9010010

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Unmanned aerial vehicle (UAV) navigation technology has made notable strides, leading to increased focus from experts in robotics and artificial intelligence. UAVs provide versatile, economical, and efficient solutions for a range of applications. These applications include precision agriculture [1,2], search and rescue operations [3,4], and wildlife conservation [5]. Reliable collision avoidance systems are essential for effective teamwork among multiple UAVs, enabling each UAV to determine the most efficient route from origin to

destination while avoiding obstacles. Consequently, UAV collision avoidance remains a critical challenge and a key research focus.

Traditional UAV collision avoidance methods, such as centralized strategies [6], rely on ground-based control stations to interact with all UAVs, accessing detailed information and issuing directives. However, this reliance on communication limits their applicability in complex scenarios [7]. Decentralized frameworks have been developed to enable UAVs to independently determine optimal trajectories using onboard sensors [8]. While these methods represent significant advancements, they often depend on predefined models and parameters, which can limit adaptability to new environments.

Recent progress in image processing and deep learning has positioned deep reinforcement learning (DRL) as a highly promising approach for UAV navigation [9–11]. DRL utilizes deep neural networks to handle high-dimensional inputs, such as images, enabling seamless learning from raw data directly to policy decisions. This approach reduces the need for manual feature extraction [12], thereby increasing the efficiency and effectiveness of UAV navigation systems.

In a manner akin to advanced neural network techniques, deep reinforcement learning relies heavily on data. It operates under the assumption that both training and evaluation datasets are independently and identically distributed (IID). However, this premise frequently proves unrealistic in practical applications. For example, a UAV system based on DRL trained within a controlled environment might later be deployed in diverse and unforeseen settings, such as forests or mountainous terrains, where conditions can differ vastly.

In order to explore the difficulties associated with the adaptability of DRL in novel environments, we re-examine the seminal work on UAV obstacle avoidance using DRL, with a focus on SAC + RAE [12]. In our study, the SAC + RAE model was trained in a particular environment (e.g., playground), and its ability to generalize was assessed through testing across a variety of distinct environments. As depicted in Figure 1, the model showed promising results when evaluated in the same playground setting.

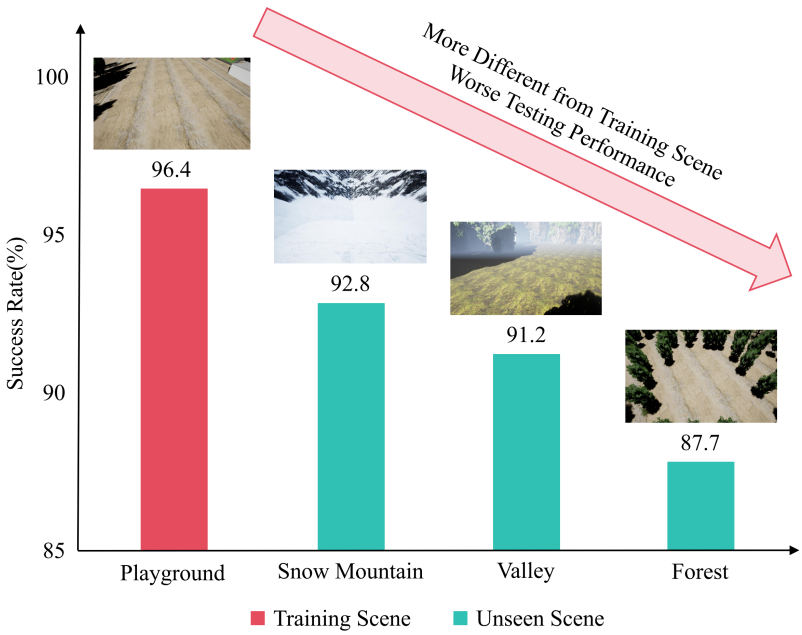


Figure 1. An illustration of a generalization capacity assessment. In the presence of novel situations, the existing DRL approach experiences significant declines in navigation success rates.

Nonetheless, its performance noticeably declined in previously unseen scenarios (e.g., snowy mountain, valley, and forest), with larger disparities between training and

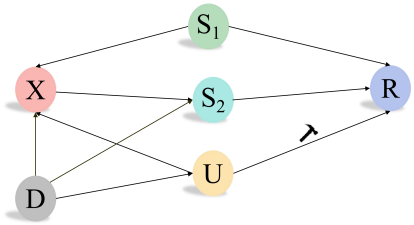
testing environments leading to poorer outcomes. These observations clearly highlight a generalization issue inherent to existing DRL-based approaches.

A key factor for the widespread adoption of UAV technologies is the creation of collision avoidance systems that can seamlessly adjust to unfamiliar environments. Efforts to enhance the generalization capabilities of DRL models in this context can be broadly classified into two main strategies: augmentation-based [13–15] and regularization-based [16–19] approaches. Augmentation-based techniques focus on modifying the training environments using image augmentation methods such as adding noise, color jittering, and blurring to help the DRL model adapt to environmental changes during deployment. On the other hand, regularization-based methods impose constraints during model training—such as encouraging sparsity or smoothness in model weights—to boost generalization and reduce overfitting. However, these manually crafted strategies are inherently limited in their breadth and fail to comprehensively address the wide array of potential deployment environments, resulting in suboptimal performance when applied to unseen settings.

To address the challenge of model generalization across different environments, we conducted a comprehensive analysis of the SAC + RAE architecture. Our analysis revealed that the limited generalization capabilities of DRL algorithms may be attributed to unstable and unreliable visual representations, such as fluctuating feature activations and inconsistent image encodings. SAC + RAE employs a regularized auto-encoder (RAE) [20] to derive visual features from images captured via UAVs. However, it includes all visual elements, regardless of their importance to collision avoidance. Irrelevant background characteristics unique to the training setting may lead to misleading associations between sensory inputs and the actions they trigger. These unintended links can undermine the task's integrity by causing the model to rely on extraneous environmental cues, rather than focusing on the pertinent aspects of the task itself [21–23].

Visual representations often include task-irrelevant factors that can hinder policy learning, making the generalization capability of DRL dependent on identifying and removing these factors. Causal representation learning [24,25] offers a viable solution by distinguishing and isolating causal factors essential for UAV navigation. Causal factors provide meaningful information necessary for navigation tasks, enabling DRL models to understand underlying causal structures across diverse scenarios and thereby enhance their ability to generalize to unseen data and environments.

In this study, we initiate the construction of an SCM to delineate the causal relationships essential to representation learning, as depicted in Figure 2. We hypothesize that only the causal factors S (e.g., obstacle features) impact the representation learning process. We introduce the causal representation disentanglement (CRD) approach to effectively pinpoint and separate non-causal factors, such as background features. In particular, we design a background intervention component that introduces modifications to background features and leverages multiple supervisory signals, including consistency loss and task-specific loss, to disentangle sub-features. This method enables the distinction between causal and non-causal factors, which enhances task efficiency. Subsequently, only the causal sub-features are utilized in policy learning, while non-causal factors are explicitly excluded, markedly enhancing the generalization ability of DRL in unfamiliar scenarios. Additionally, we designed a simulation environment to assess the collision avoidance capabilities of UAVs. The training and testing scenarios are distinct, allowing for an effective evaluation of the generalization capabilities of DRL models. Comprehensive experiments demonstrate the superiority and efficacy of our proposed approach, thereby strongly supporting the continued development and implementation of UAV systems.



X: Image
R: Representation
D:Domain Information
S_1: Causal Domain-Invariant factor
S_2: Causal Domain-Specific factor
U: Non-causal Factors

Figure 2. The process of representation learning involves the use of a structural causal model (SCM).In the context of representation learning, an SCM is utilized. An image, denoted as X, consists of causal elements S, which include both domain-invariant causal factors S_1 and domain-specific causal factors S_2, as well as non-causal elements U. However, it is solely the causal elements S that influence the representation learning process.

The contributions of our work are summarized as follows:

- We are the first to address the generalization issues in UAV collision avoidance systems that utilize DRL by adopting a causality-based perspective.
- We develop a framework for disentangling causal representations, which promotes the creation of robust and causally grounded visual features. Our approach significantly enhances the generalization capability of DRL models in new environments by intentionally eliminating non-causal factors.
- We carry out extensive experiments and thorough analyses on tasks involving collision avoidance among multiple UAVs. These evaluations confirm the superior performance and effectiveness of our proposed methodology.

2. Related Work

2.1. UAV Collision Avoidance

Traditional UAV obstacle avoidance methods can be broadly categorized into three types: geometric-based methods [26–28], optimization-based methods [29,30], and sensor-based methods [31,32]. Geometric-based methods achieve obstacle avoidance by analyzing the geometric relationships between UAVs and obstacles. For example, Goss et al. [26] employed a mixed geometric and collision cone approach to resolve conflicts or collisions between two aircraft in a 3D environment using current positions and velocities. Seo et al. [27] extended the collision cone method to allow UAV swarms to avoid obstacles. Lin et al. [28] proposed an efficient 3D collision avoidance algorithm that combines geometric avoidance with the selection of critical avoidance start times to quickly navigate multiple obstacles. Optimization-based methods plan optimal obstacle avoidance paths using geographic information and probabilistic search techniques to improve efficiency. Pérez-Carabaza et al. [29] introduced a minimum time search (MTS) planner based on ant colony optimization, which optimizes UAV search trajectories while considering communication and collision avoidance constraints. Biswas et al. [30] developed a path planning method based on particle swarm optimization for autonomous systems in dynamic environments. Sensor-based methods use onboard sensors to detect and avoid obstacles, enabling rapid response and path adjustments for drones or robots in dynamic settings. Zsedrovits et al. [31] proposed a visual aircraft detection algorithm for drones, effective under both clear and cloudy conditions. Liang et al. [32] introduced a UAV obstacle avoidance method composed of three modules: environmental perception, algorithmic avoidance, and motion control, allowing for safe navigation in low-altitude complex environments. However, geometric-based methods can suffer from high computational complexity in intricate environments, which may impair real-time performance. Optimization-based

methods, while effective in static environments, require detailed environmental information and pre-planning, limiting their adaptability to dynamic changes. Sensor-based methods, though flexible and free from pre-planning, may get stuck in local minima in complex environments, necessitating integration with more efficient path planning algorithms to ensure the UAVs reach their target.

Deep reinforcement learning (DRL) methods have achieved considerable success in multi-agent collision avoidance by formulating decision-making strategies directly from sensory inputs, thereby eliminating the need for prior information. Within the established framework, Qie et al. [33] employed the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm to address challenges associated with planning trajectories and allocating targets for multiple unmanned aerial vehicles. Additionally, Xue et al. [34] presented the Multi-Agent Recursive Deterministic Policy Gradient (MARDPG) method. This approach extends deep deterministic policy gradients to effectively manage multiple UAVs. This advancement further demonstrates the effectiveness of deep learning techniques in solving complex navigation challenges. In addition, Fei and associates [35] introduced an enhanced deep reinforcement learning model named FRDDM-DQN. This model combines Faster R-CNN with a data storage system, enabling UAVs to autonomously navigate and avoid collisions even in environments lacking radar and experiencing communication disruptions. Similarly, Huang et al. [12] created a vision-based, decentralized strategy for collision avoidance using deep reinforcement learning. This strategy facilitates scalable and resilient path planning without the need for communication, representing the inaugural application of this functionality in systems involving multiple UAVs. Nevertheless, current DRL methodologies frequently encounter difficulties in accurately representing essential task-specific features due to inherent limitations in feature representation. This shortfall consequently hampers the ability of these models to generalize effectively across different scenarios.

2.2. Robust Representation Learning

The endeavor to create robust feature representations is a dynamic research domain within computer vision and machine learning. By aiming to comprehend and depict essential explanatory factors within an efficient framework, feature disentanglement strives to establish a cohesive latent feature space. Within this hidden space, specific dimensional features are crafted to encode distinct semantic features [36].

Additionally, Liu et al. [37] introduced the Unified Feature Disentanglement Network (UFDN), utilizing adversarial training techniques to ensure uniform feature representations across various source domains. Conversely, Peng et al. [38] developed the Deep Adversarial Disentanglement Auto-Encoder (DADA), an innovative model capable of effectively separating features using a single source domain in conjunction with multiple unannotated target domains.

Nevertheless, these approaches primarily focus on surface-level associations and neglect to examine underlying causal links, thereby compromising their reliability. Addressing this drawback, Cai et al. [39] developed a method that divides features into mutable and stable components through the use of a VAE. Furthermore, Kong et al. [40] tackled the problem by focusing on data generation models, emphasizing the importance of maintaining minimal alterations in causal mechanisms across various domains. Their study demonstrated that latent variables attain partial identifiability under the fulfillment of specific conditions. These prerequisites involve having an adequate number of domains, subjecting latent variables to monotonic transformations, and guaranteeing a homogenous distribution of labels. Simultaneously, Li et al. [25] attained the full identifiability of latent

variables by employing causal generative processes while relying on fewer assumptions related to domains.

Motivated by these studies, we tackle the generalization challenge in UAV collision avoidance based on DRL through the development of a causal representation disentanglement method. This method systematically eliminates non-causal elements from visual data, thereby improving the model's capacity to generalize effectively to novel environments.

3. Approach

3.1. System Model and Problem Formulation

Our method builds on the foundation established by SAC + RAE [12]. It integrates centralized training with a distributed execution framework to improve policy optimization. In multi-UAV collision avoidance, each drone operates autonomously. It relies exclusively on its onboard sensors—a forward-facing camera and an inertial measurement unit (IMU)—to navigate and avoid obstacles. This configuration leads to a partially observable environment. We formally define this scenario as a decentralized, partially observable Markov decision process (Dec-POMDP).

3.1.1. Observation Space

Every drone collects data composed of three unique elements: the encoded depth image, o^t, determine the target's specific location based on the coordinate system established by the UAV itself, o^g, and the UAV's flight speed, o^v.

3.1.2. Action Space

We use a continuous action space framework to improve UAV flight stability and flexibility. The action vector a is generated by the policy network $\pi(s)$ and is defined as $a = [v_x^{cmd}, v_z^{cmd}, v_\omega^{cmd}]$. Based on this:

- v_x^{cmd} specifies the UAV's forward velocity along its longitudinal axis.
- v_z^{cmd} indicates the vertical ascent rate, representing either upward or downward movement.
- v_ω^{cmd} control the rotational speed to manage the UAV's turning.

3.1.3. Overview of AirSim's UAV Control System Architecture

The experiments in this study were based on AirSim's UAV control system architecture, which employs a modular design to facilitate an efficient and flexible control process [41].

The action vector $\mathbf{a} = [v_x^{cmd}, v_z^{cmd}, \omega^{cmd}]$ is generated via the policy network $\pi(s)$, where each component is influenced by the forces generated via the rotors, as depicted in Figure 3.

The UAV model in AirSim is grounded in classical quadrotor dynamics, described by the following equations:

$$\begin{cases} m \cdot \frac{d\mathbf{v}}{dt} = \mathbf{F} - m \cdot \mathbf{g} \\ I \cdot \frac{d\boldsymbol{\omega}}{dt} = \boldsymbol{\tau} \end{cases} \tag{1}$$

where m represents the mass of the UAV, $\mathbf{v}$ is the velocity vector, $\mathbf{F}$ denotes the total external force acting on the UAV, $\mathbf{g}$ is the gravitational acceleration vector, I is the inertia matrix, $\boldsymbol{\omega}$ is the angular velocity vector, and $\boldsymbol{\tau}$ is the total external torque.

The control of the UAV's attitude and position is executed using the built-in proportional–integral–derivative (PID) controller in AirSim. The PID coefficients used in our simulations are summarized in Table 1, and the UAV specifications are listed in Table 2.

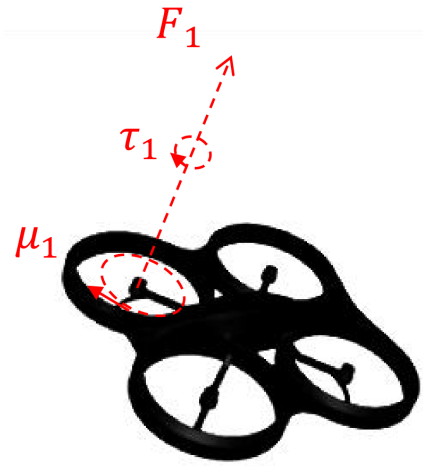


Figure 3. The quadrotor model is depicted with four vertices, each experiencing a control input, u_1. This input drives the rotational speed of the propellers, resulting in a force, F_1, and a torque, τ_1.

Through this control mechanism, the UAV can achieve stable and precise flight operations. The PID controller modulates the UAV's actuators (e.g., motor speeds) based on these computations, ensuring the precise and stable execution of velocity commands.

This architectural design effectively decouples high-level path planning and obstacle avoidance algorithms from low-level dynamics and control logic. The policy network is tasked with generating efficient velocity commands, while the physical engine and PID controller in AirSim manage the specific dynamic responses and control execution. This design not only simplifies the control process but also enhances the system's adaptability and real-time responsiveness across diverse environments.

The architecture supports the system's ability to maintain high levels of stability and reliability in dynamic environments while also accommodating the expansion and adaptation of UAVs in various deployment scenarios.

Table 1. PID coefficients for different control modes.

Controller Type	K_P	K_I	K_D
Linear-velocity PID controller	0.2	2.0	0.0
Angular-velocity PID controller	0.25	0.0	0.0

Table 2. UAV specifications.

Specification	Value
Mass	0.8 kg
Rotor count	4
Minimum motor output	0
Maximum motor output	1
Minimum angling throttle	0.05
Maximum angular rate	2.5 rad/s
Body box dimensions (m)	$0.2413 \times 0.1143 \times 0.0762$

3.1.4. Reward Function

The careful selection of a reward function is essential for providing meaningful and constructive feedback to each agent within the system. Our proposed reward structure consists of two fundamental components:

- r_g: This component guides the UAV toward its specified target. By incentivizing movement in the direction of the intended objective, r_g ensures that each UAV steadily progresses toward reaching its designated location.
- r_c: This element encourages the UAV to steer clear of obstacles. By penalizing interactions with potential hazards, r_c promotes safer navigation, thereby reducing the likelihood of collisions.

Together, these two rewards establish a balanced framework that simultaneously advances the UAVs toward their goals while ensuring safe operation within their environment.

$$r = r_g + r_c \tag{2}$$

$$r_g = \begin{cases} r_{arrival}, & \text{if } d_t < 0.5 \\ \alpha_{goal} \cdot (d_t - d_{t-1}), & \text{otherwise} \end{cases} \tag{3}$$

where d_t between the unmanned aerial vehicle and the target at the time t.

$$r_c = \begin{cases} r_{collision}, & \text{if crashed} \\ \alpha_{avoid} \cdot max(d_{safe} - d_{min}, 0), & \text{otherwise} \end{cases} \tag{4}$$

During the training phase, we assign the following values: $r_{\text{arrival}} = 50$, $r_{\text{collision}} = -10$, $\alpha_{\text{goal}} = 3$, $\alpha_{\text{avoid}} = -0.05$, and $d_{\text{safe}} = 5$.

3.2. Overview

Building upon the foundational work presented by SAC + RAE [12], our approach conducts centralized training with multiple UAVs and performs distributed execution during testing to optimize our policy, as illustrated in Figure 4.

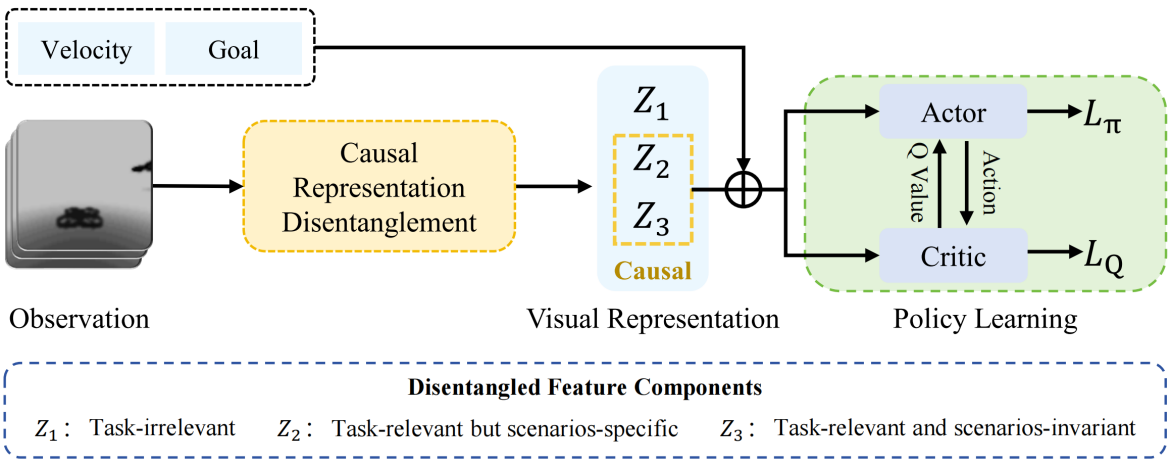


Figure 4. The architecture of our UAV collision avoidance framework.Our framework is founded on the SAC framework, which processes inputs including depth images, the UAV's current speed, and the relative position of the target goal. Based on these inputs, it generates the necessary flight control commands. To improve the extraction of visual features, we introduce a method for disentangling causal representations. This approach detects and separates the essential causal elements, ensuring that only these components are forwarded to the next stage of policy learning.

In this work, our objective is to optimize the representation extraction process by implementing causal representation disentanglement, which allows us to identify and isolate hidden causal factors while eliminating task-irrelevant (non-causal) elements. This strategy substantially improves the generalization ability of the deep reinforcement learning (DRL) model.

Inspired by previous research on feature disentanglement [25,40,42], we decompose the extracted representations into three distinct categories:

- Task-irrelevant components, $Z_1 \in \mathbb{R}^{n_1}$;
- Task-relevant but scenario-specific components, $Z_2 \in \mathbb{R}^{n_2}$;
- Task-relevant and scenario-invariant components, $Z_3 \in \mathbb{R}^{n_3}$.

Specifically, Z_2 (e.g., obstacle distribution) and Z_3 (e.g., the distance between the drone and an obstacle) serve as causal elements that supply crucial information for the collision avoidance task. Conversely, Z_1 (e.g., background patterns) is irrelevant to the task and may foster misleading correlations during data-driven training, thereby impeding the generalization of the UAV system in new environments. By employing the proposed causal representation disentanglement method, we successfully isolate these three components. We then transfer only Z_2 and Z_3 to the subsequent policy learning phase, effectively eliminating the influence of Z_1.

3.3. Causal Identifiability Analysis

In this part, we analyze the causal identifiability of the proposed causal latent model. We begin by demonstrating that achieving identifiability for the proposed causal model is unattainable without additional domain information. We then prove that incorporating such domain information allows the identification of non-causal features from depth image data. This proof of identifiability offers a solid theoretical foundation for our algorithm design and ensures the rationality of the proposed method.

While the framework of autoencoders enables efficient learning of deep latent-variable models, a common challenge is the lack of identifiability guarantees. Khemakhem et al. [43] demonstrate that any model with an unconditional latent distribution is unidentifiable. Specifically, they show, from the perspective of factor priors and nonlinear independent component analysis (ICA), that without external variables, it may be impossible to recover the true latent variables solely from observed data.

Similarly, in this paper, the absence of additional domain information results in causal and non-causal features being equivalently represented in the observed depth image data, making it difficult for autoencoders to differentiate between them. However, Tian et al. [44] show that utilizing multiple source domains with varying data distributions can help identify causal structures, as causal knowledge can be inferred from distributional changes. Therefore, incorporating additional domain information is crucial for accurately identifying non-causal features.

Following the SCM depicted in Figure 2, we adopt the subsequent assumptions:

- Continuous and strictly positive density: The latent variables are governed by a probability density function that is both continuous and strictly greater than zero across the domains Z, D, and Y. Formally, this condition is expressed as $p_{z|d,y} > 0$ for all $z \in Z$, $d \in D$, and $y \in Y$. In this framework, Y denotes the label, which encapsulates the impact of the reward function on the latent variables.
- Conditional independence: conditioned on d and y, each z_i is independent of any other z_j for $i, j \in \{1, \cdots, n\}, i \neq j$, i.e., $\log p_{z|d,y}(z|d,y) = \sum_{i=1}^{n} \log p_{z_i|d,y}(z_i|d,y)$.
- Linear independence: for any $z \in Z$, there exist $n_1 + n_2 + 1$ combinations of (d, y), such that these $n_1 + n_2$ vectors $v(z, d_j, y_c) - v(z, d_0, y_0)$ are linearly independent, where vector $v(z, d, y)$ is defined as follows:

$$v(z, d_j, y) = \left(\frac{\partial \log p_{z_1|d,y}(z_1|d,y)}{\partial z_1}, \cdots, \frac{\partial \log p_{z_n|d,y}(z_n|d,y)}{\partial z_n} \right) \tag{5}$$

Through the modeling of the aforementioned assumptions, z_1, z_2, and z_3 are subspace-identifiable.

First, we construct an invertible transformation, h, between the ground-truth z and estimated $\hat{z}$. Then, we consider changes in the domain and labels to construct a full-rank system of linear equations, which has a unique solution, $\frac{\partial z_i}{\partial \hat{z}_j} = 0$. Because the Jacobian matrix of h is non-singular, for every z_i, where $i \in \{1, \cdots, n\}$, a corresponding h_i exists, such that $z_i = h(\hat{z}_i)$.

3.4. Causal Representation Disentanglement

Achieving causal representation disentanglement fundamentally involves directing the representation elements, specifically Z_1, Z_2, and Z_3, to acquire distinct semantic concepts, as intended. The methodology of causal representation disentanglement is depicted in Algorithm 1. In preceding successful research on feature disentanglement [39,45], scholars frequently employed data from diverse sources to reveal invariant underlying mechanisms to distinguish the causal representation components. Aligning with these strategies, we implement a straightforward yet effective technique known as background intervention to generate multi-domain data for this objective. Additionally, we develop multiple critical loss criteria to facilitate and direct the separation of the constituent representation components.

Algorithm 1 Causal representation disentanglement with training loop

Input: Image X, Encoder E_ϕ, Decoder D_ψ, Weights λ_{align}, λ_{force}, Learning rate η
Output: Causal representations Z_{causal}

```
Initialize parameters φ, ψ;
while not converged do
    Z ← E_φ(X)
    Z_1, Z_2, Z_3 ← Disentangle(Z)
    Z_causal ← [Z_2, Z_3]
    X̂ ← D_ψ(Z)
    L_rec ← ||X − X̂||^2
    L_align ← Compute for Z_3
    L_force ← Penalize Z_1
    L ← L_rec + λ_align L_align + λ_force L_force
    φ ← φ − η∇_φ L
    ψ ← ψ − η∇_ψ L
end while
```

3.4.1. Background Intervention

Our analytical experiment (Figure 1) and the developed structural causal model (SCM) (Figure 2) reveal that non-causal elements, such as background patterns, can introduce deceptive information that negatively affects the model's ability to generalize to unfamiliar scenarios within the collision avoidance framework. To facilitate the network in identifying latent causal factors, we apply perturbations to these non-causal elements through the implementation of a background intervention strategy, thereby enabling the generation of multi-domain data. Specifically, we employ various commonly used data augmentation techniques, including the addition of random noise, motion blurring, and Gaussian blurring, to carry out these interventions. It is important to highlight that these augmentation methods exclusively alter background information, leaving task-related execution unchanged. Consequently, it can be reasonably inferred that the introduced perturbations are confined to non-causal components.

3.4.2. Loss Function

As shown in Figure 5, we utilize VAE [46] to separate causal features from non-causal features. The encoder uses $q(z|x)$ to transform the input image into the latent representation space Z, while the decoder $q(x|z)$ reconstructs this learned representation back into an image resembling the original input. Background intervention primarily perturbs Z_1, forcing the model to learn to encode background information solely in Z_1 and thereby achieving the decoupling of causal and non-causal features.

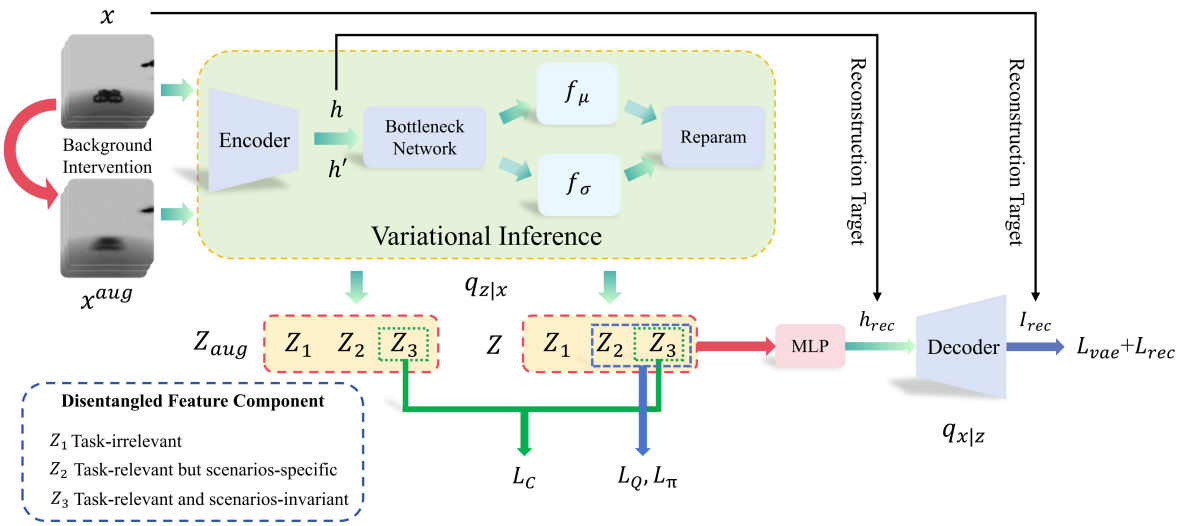


Figure 5. Causal representation disentanglement illustration. We utilize a variational auto-encoder to extract features and subsequently design specialized loss functions that guide predefined components to develop distinct semantic meanings. The term 'Reparam' denotes the reparameterization technique used in this process.

Additionally, to ensure the accuracy of separating causal and non-causal features, we designed multiple loss functions. First, we use a reconstruction loss to measure the difference between the image generated via the decoder and the original input image.

$$\mathcal{L}_{vae} = D_{KL}[q(z|x)||p(z)] - \mathbb{E}_{q(z|x)}[\log q(x|z)] \tag{6}$$

where $p(z)$ denotes a prior probability distribution. To enhance the bottleneck's network architecture and amplify model diversity, we redesigned h:

$$\mathcal{L}_{rec} = (h - h_{rec})^2 \tag{7}$$

Viewed causally, the background disturbance process represents a causal intervention on images [47], as shown in Figure 5. This intervention aids the model in learning how to disentangle the causal factors within the images. Additionally, we developed an alignment loss to ensure that perturbations to the non-causal representation Z_1 do not affect the causal representations Z_2 and Z_3, as detailed below:

$$\mathcal{L}_C = \frac{1}{C}\sum_{i=1}^{C} ||z_3 - z^i_{3,aug}||^2 \tag{8}$$

where C indicates how many data augmentation techniques are employed, and $z^i_{3,aug}$ signifies the features derived from the ith technique.

Additionally, to guarantee that causal elements are relevant to the task and provide crucial information for policy learning, the sub-features $o = [Z_2, Z_3]$ are forwarded to the

next policy learning phase. As a result, the SAC model's learning process can be generally split into two main steps.

The initial phase encompasses assessing the policy, which seeks to precisely estimate the Q-function:

$$\mathcal{L}_Q = \mathbb{E}_{(o,o')\sim q(z|x),(a,r)\sim\mathcal{B}}\left[\left(Q(o,a) - r - \gamma\bar{V}(o')\right)^2\right] \tag{9}$$

During each iteration, a pair (o, o') is obtained from the encoder $q(z|x)$, and a tuple (a, r) is retrieved from the replay buffer $\mathcal{B}$.

$$\bar{V}(o) = \mathbb{E}_{a\sim\pi_\theta}[\bar{Q}(o,a) - \alpha \log \pi(a|o)] \tag{10}$$

where the objective Q-function is represented as $\bar{Q}$.

The subsequent stage encompasses refining the policy, aiming to revise the model's strategy:

$$\mathcal{L}_\pi = \mathbb{E}_{o\sim q(z|x)}[D_{KL}(\pi(\cdot|o)\|\mathcal{Q}(o,\cdot))] \tag{11}$$

where $\mathcal{Q}(o,\cdot) \propto \exp\left\{\frac{1}{\alpha}Q(o,\cdot)\right\}$.

It is important to emphasize that only Z_2 and Z_3 are exclusively provided to the reinforcement learning model, thereby nullifying the impact of Z_1.

4. Experiment

4.1. Experimental Setup and Parameter Configuration

We constructed various simulation environments by integrating Unreal Engine (UE) with the AirSim simulator [41]. These simulation experiments were conducted on a workstation running Ubuntu 20.04, equipped with an Intel i9-13900K processor and an NVIDIA GeForce RTX 4090 GPU. The hyperparameters employed in this study are thoroughly outlined in Table 3. In addition, the specific details of our experiment can be found in the video included in our Supplementary Materials.

Table 3. Hyperparameters for policy training.

Parameter Name	Value
Batch size	128
Max episodes	300
Update times	400
Replay buffer $\mathcal{B}$ capacity	20,000
Discount γ	0.99
Encoder learning rate	10^{-4}
Critic's learning rate	10^{-4}
Critic's target update frequency	2
Critic's Q-function soft-update rate τ_Q	0.01
Critic's encoder soft-update rate τ_{enc}	0.05
Actor's learning rate	10^{-4}
Actor's update frequency	2
Actor's log stddev bounds	$[-10, 2]$
Optimizer	Adam
n_1	8
n_2	6
n_3	42

4.2. Performance Metrics and Experimental Scenarios

4.2.1. Performance Metrics

Following the methodology of SAC + RAE [12], we utilized the subsequent evaluation metrics to gauge performance across diverse scenarios:

- Swarm success rate (SSR): The swarm success rate quantifies the fraction of trials where the entire UAV fleet successfully reaches the designated target locations. This metric is particularly significant in real-world applications necessitating coordinated efforts among multiple UAVs, as a mission is typically deemed successful only when every UAV accomplishes its respective objective.
- Individual success rate (ISR): The individual success rate assesses the proportion of UAVs that attain their assigned target positions without encountering any collisions within a defined time period. This metric highlights each UAV's autonomous performance and dependability throughout the mission.
- Success weighted by path length (SPL): This indicator assesses whether a UAV successfully reaches its intended destination by following the most optimal route. It is determined using the following formula:

$$SPL = \frac{\sigma \cdot l}{\max(l, p)} \tag{12}$$

  In this context, l signifies the shortest distance from the UAV's starting location to the target, p indicates the actual distance covered by the UAV, and σ is a binary variable that shows whether the mission was successfully accomplished.
- Extra distance: This metric quantifies the average additional distance covered by UAVs compared to the shortest possible path from their initial positions to their designated targets. Both the mean and standard deviation are provided for this measurement.
- Average speed: This measure calculates the mean velocity of all UAVs during the testing phase. As with extra distance, this metric also reports both the average and standard deviation values.

4.2.2. Experimental Scenarios

To assess the generalization capabilities of DRL models, we developed two separate scenarios. One scenario is used for training, while the other is utilized for evaluation, as illustrated in Figure 6.

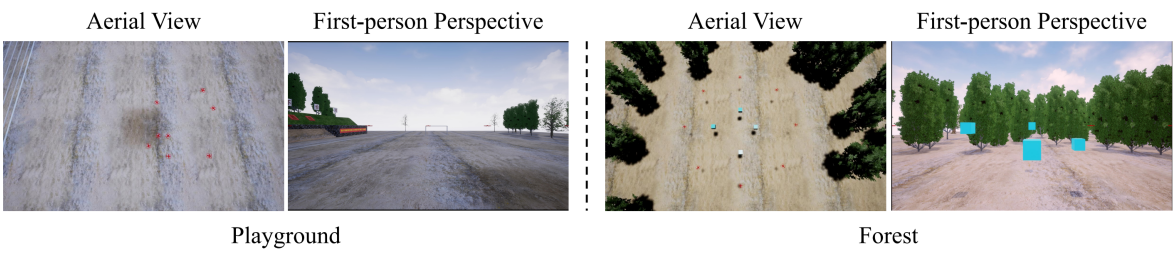


Figure 6. Scenarios for model training and evaluation. The model is trained using the playground scenario, whereas the testing phase employs the forest scenario, which includes previously unseen obstacles. To enhance clarity, both aerial viewpoints and first-person perspectives are provided. For optimal viewing quality, zooming in is recommended.

The training environment is a playground without any obstacles. In each training cycle, the starting and target positions for each drone are randomly assigned within a cubic area that is 14 m long, 14 m wide, and 5 m high. This method aids drones in improving the adaptability of their strategies across diverse environments, as they must navigate through changing conditions.

Conversely, for testing purposes—and taking into account practical UAV applications such as geological surveys [48] and search and rescue missions [4]—we created a forested setting that incorporates several previously unseen obstacles to test the DRL models' capacity to generalize. Furthermore, during each testing phase, all drones are initially positioned uniformly around a circle with a 16 m radius at a designated altitude, with their target locations located on the opposite side of the circle. This commonly used and more demanding testing approach more effectively validates the acquired collision avoidance strategies.

4.3. Performance Comparison

In this work, we focus on improving the generalization capability of deep reinforcement learning (DRL) models. As our baseline, we adopt the previously state-of-the-art DRL-based method SAC + RAE [12], despite the existence of several traditional approaches [49,50], such as probabilistic and deterministic algorithms, which can also address collision avoidance in multi-UAV scenarios. To evaluate our proposed CRD module, we integrated it into the baseline model and compared its performance with other established methods designed to address generalization issues. These include augmentation-based approaches, such as AutoAugment [15] and DrAC [14], as well as regularization-based techniques, including $L1$ norm [16,51] and $L2$ norm [17,52]. Additionally, we compared our method with the latest CRL approach [53], which is specifically designed to improve collision avoidance in multi-UAV scenarios.

For a fair comparison, we followed the official implementations of each method and conducted uniform model training within predefined training scenarios. Specifically, all methods were trained under identical simulation environments and parameter settings. During training, we ensured that each method underwent sufficient iterations to achieve convergence and optimal results. Finally, the trained models were evaluated in the same testing scenarios to ensure consistency and fairness in the performance comparison.

The experimental results are summarized in Table 4. From the results, we can draw the following observations. First, our proposed CRD demonstrates significant improvements in both the navigation success rate and SPL metrics, confirming the superiority and efficiency of our method. Second, although CRD results in a slightly longer planned flight path and a slower speed, this is due to the UAVs performing more collision avoidance maneuvers, which ultimately leads to a higher success rate.

Table 4. Performance comparison of various methods in the testing scenario with an unseen background and obstacles, using cycle pattern initialization.

Method	SSR (%)	ISR (%)	SPL (%)	Extra Distance (m)	Average Speed (m/s)
SAC + RAE	29.6	87.7	82.2	10.591/4.443	0.457/0.090
+ AutoAugment [15]	53.4	90.6	82.7	3.192/0.840	0.854/0.154
+ DrAC [14]	50.0	88.7	77.4	4.712/1.547	1.026/0.190
+ $L1$ Norm [16,51]	69.6	92.9	84.2	2.459/0.855	0.870/0.107
+ $L2$ Norm [17,52]	58.4	93.1	79.8	4.410/1.522	0.912/0.113
+ CRL [53]	71.1	94.5	85.1	3.924/0.883	0.844/0.097
+ Our CRD	**93.6**	**98.9**	87.3	3.728/0.677	0.812/0.086

4.4. Ablation Study

In this subsection, we describe experiments performed to demonstrate the efficacy of our proposed approach.

4.4.1. Representation Components

The fundamental strategy to enhance generalization in our approach involves decomposing representations into three distinct components (i.e., Z_1, Z_2, and Z_3) and exclusively utilizing the task-relevant components (Z_2 and Z_3) for subsequent policy learning. This methodology deliberately removes the influence of the task-irrelevant component (Z_1).

To gain deeper insights into the role of each representation component, we performed an ablation study examining various combinations of these components. As presented in Table 5, two primary findings are evident.

Table 5. Ablation study on different combinations of representation components.

Z_1	Z_2	Z_3	SSR (%)	ISR (%)
	✓	✓	**93.6**	**98.9**
	✓		8.9	71.6
		✓	6.4	67.3
✓	✓	✓	11.7	78.4

Firstly, both Z_2 and Z_3 are essential for achieving optimal policy learning, and their combined use results in the highest performance. Secondly, the inclusion of Z_1 diminishes generalization capabilities, as it encompasses task-irrelevant information that can readily introduce spurious correlations, thereby causing erroneous action predictions.

4.4.2. Scalability

To evaluate the generalization and scalability of our method, we separately increased the number of UAVs and obstacles during the testing phase. As illustrated in Table 6, our proposed CRD consistently surpasses the baseline model (SAC + RAE) under various conditions. These experimental findings indicate that our method effectively generalizes to a diverse set of unseen scenarios, which is essential for real-world applications.

Table 6. Scalability analysis experiments. Comparisons were conducted under varying numbers of UAVs and obstacles. ↑ means an increase in percentage points.

	Num	SAC + RAE		+ Our CRD	
		SSR (%)	ISR (%)	SSR (%)	ISR (%)
Obstacle	4	29.6	87.7	**93.6** (↑ 64.0)	**98.9** (↑ 11.2)
	6	22.4	82.6	**89.6** (↑ 67.2)	**97.6** (↑ 15.0)
	8	18.4	80.6	**87.0** (↑ 68.6)	**96.8** (↑ 16.2)
	10	16.6	78.9	**80.0** (↑ 63.4)	**95.7** (↑ 16.8)
UAV	8	29.6	87.7	**93.6** (↑ 64.0)	**98.9** (↑ 11.2)
	10	20.0	84.0	**91.8** (↑ 71.8)	**93.6** (↑ 9.6)
	12	11.8	82.4	**89.6** (↑ 77.8)	**91.0** (↑ 8.6)
	14	2.5	80.6	**85.4** (↑ 82.9)	**88.2** (↑ 7.6)

4.4.3. More Evaluation Scenes

To further verify the generalization capability, we designed a variety of additional scenes for evaluation, including a snowy mountain and valley, as shown in Figure 7. As indicated in Table 7, our proposed CRD consistently improves performance across different scenes, including both seen and unseen environments.

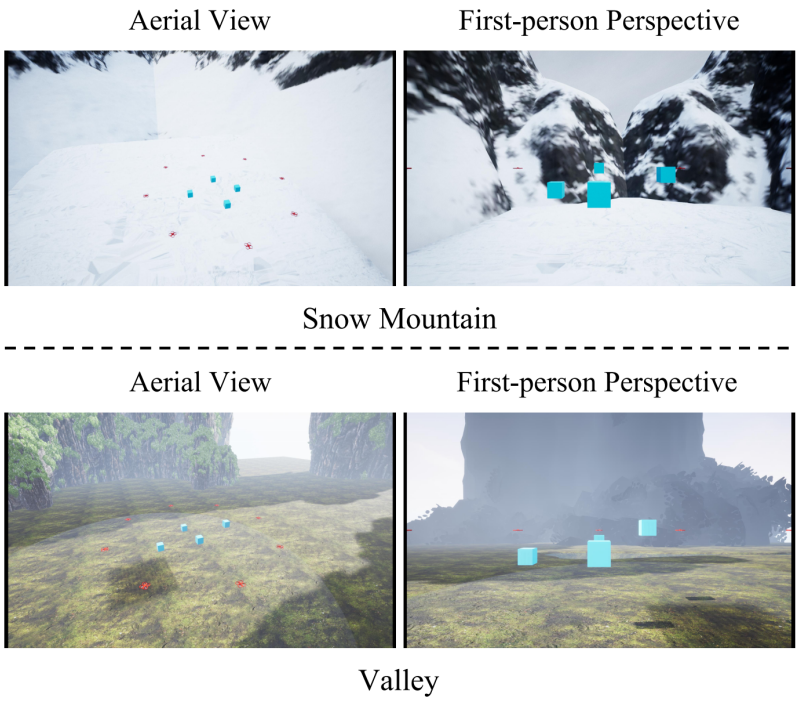


Figure 7. More testing scenes. Illustration of extra testing scenes used in Table 7.

Table 7. Performance improvement under conditions with various scenes.

Scenes	Seen/ Unseen	Method	SSR (%)	ISR (%)	SPL (%)	Extra Distance (m)	Average Speed (m/s)
Playground	Seen	Baseline	87.5	96.4	81.4	6.156/3.449	0.527/0.099
		+ Our CRD	**100.0** (↑ 12.5)	**100.0** (↑ 3.6)	**87.8** (↑ 6.4)	**4.458/0.302**	**0.852/0.038**
Forest	Unseen	Baseline	29.6	87.7	82.2	10.591/4.443	0.457/0.090
		+ Our CRD	**93.6** (↑ 64.0)	**98.9** (↑ 11.2)	**86.1** (↑ 3.9)	**3.254/0.564**	**0.812/0.086**
Snowy mountain	Unseen	Baseline	65.1	92.8	72.3	9.236/2.927	0.504/0.068
		+ Our CRD	**99.2** (↑ 34.1)	**99.8** (↑ 7.0)	**88.5** (↑ 16.2)	**3.629/0.309**	**0.967/0.061**
Valley	Unseen	Baseline	52.3	91.2	68.2	11.361/5.329	0.460/0.092
		+ Our CRD	**96.8** (↑ 44.5)	**99.8** (↑ 8.7)	**88.6** (↑ 20.4)	**3.568/0.359**	**0.931/0.056**

4.4.4. Other Disentanglement Techniques

The key component of our proposed method is a novel representation disentanglement technique, developed from a causal perspective. To further showcase the effectiveness of our approach, we compared its performance with existing representation disentanglement methods, including β-VAE [54], Factor-VAE [55], and DAVA [56]. As shown in Table 8, our proposed CRD outperforms these popular representation disentanglement techniques, verifying its effectiveness.

Table 8. Performance comparison with other feature disentanglement techniques.

Method	SSR (%)	ISR (%)	SPL (%)	Extra Distance (m)	Average Speed (m/s)
SAC + RAE	29.6	87.7	82.2	10.591/4.443	0.457/0.090
+ β-VAE [54]	50.6	88.3	84.4	1.335/0.932	1.211/0.068
+ Factor-VAE [55]	51.4	88.7	83.5	2.032/1.324	1.179/0.084
+ DAVA [56]	46.4	87.9	82.0	2.138/1.557	1.124/0.069
+ Our CRD	**93.6**	**98.9**	**86.1**	3.254/0.564	0.812/0.086

4.4.5. Trajectory Visualization

Additionally, we visualized the UAV flight trajectories to intuitively assess flight quality, as depicted in Figure 8. The trajectories are presented both as a perspective view and a three-view drawing, with each UAV's trajectory represented in a distinct color. Our method generates smoother and more complete trajectories. In contrast, SAC + RAE results in collisions between UAVs, underscoring the superior robustness and effectiveness of our approach in path planning.

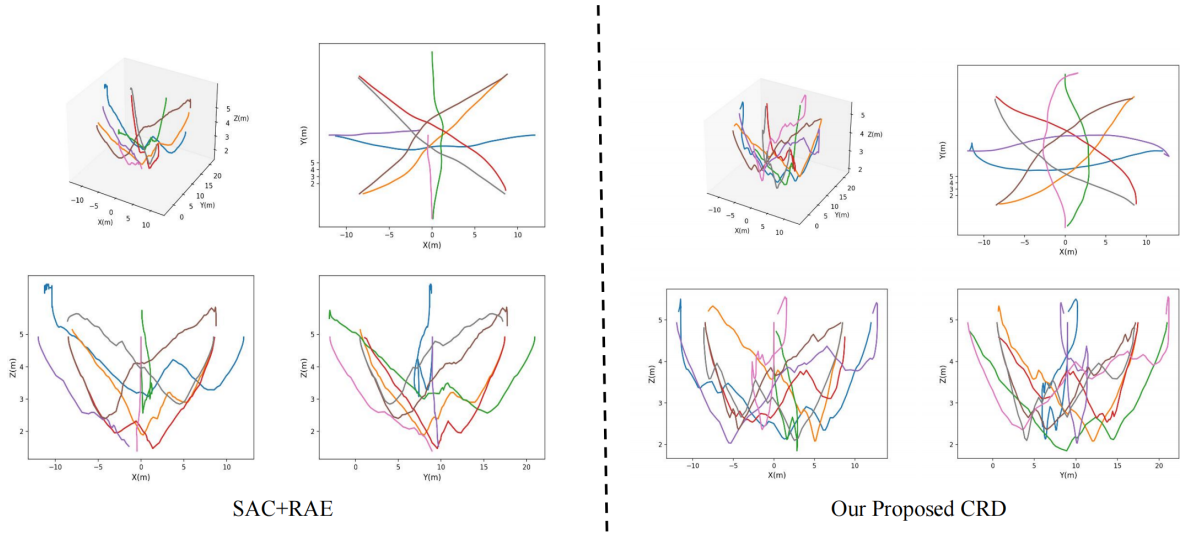


Figure 8. Visualizing UAV trajectories with perspective and three-view illustrations. Different colors are used to represent the flight paths of various UAVs. Viewing the figures in color provides the best clarity.

5. Conclusions

In this work, we have explored the generalization difficulties that encounter a modern DRL-based multi-UAV collision avoidance framework when it is applied to new environments. To tackle this challenge, we presented a novel representation learning approach, specifically causal representation disentanglement, which divides visual representations into several distinct components, each with unique semantic meanings, and utilizes only the causal elements for subsequent policy development. This strategy effectively removes the influence of task-irrelevant factors, thereby enhancing the model's generalization capabilities. Extensive experiments conducted across a variety of unseen testing environments demonstrate the practicality and effectiveness of our proposed method, outperforming existing state-of-the-art techniques.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/drones9010010/s1, Video S1: video-of-crd.mp4.

Author Contributions: Conceptualization, Z.F.; methodology, Z.X. and Z.F.; software, C.L. and Z.F.; validation, C.L. and Z.X.; formal analysis, Z.F., C.L. and Z.X.; investigation, Z.F., C.L., G.H. and Z.X.; resources, Z.F. and J.Z.; data curation, Z.F., C.L. and Z.X.; writing—original draft preparation, Z.F., C.L. and Z.X.; writing—review and editing, Z.F., C.L. and Z.X.; visualization, Z.F., C.L. and Z.X.; supervision, W.L., D.W., Y.C., Z.H., R.C., J.Z., G.H. and Z.F.; project administration, J.Z. and Z.F.; funding acquisition, J.Z., W.L. and Z.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Science and Technology Major Project (grant numbers 2021ZD0111502), the National Natural Science Foundation of China (grant numbers 62176147, 51907112, U2066212, 61961036, 62162054, 62406186, and 62476163), the Science and Technology Planning Project of Guangdong Province of China (grant numbers 2023A1515011574, 2022A1515110566, and 2022A1515110660), the STU Scientific Research Foundation for Talents (grant numbers NTF21001, NTF21052, and NTF22030), and the Key Scientific Research Project of Universities in Guangdong Province, China (grant number 2024ZDZX1021).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DRL	Deep Reinforcement Learning
UAV	Unmanned Aerial Vehicle
IID	Independently and Identically Distributed
SCM	Structural Causal Model
RAE	Regularized Auto-Encoder
CRD	Causal Representation Disentanglement
POMDP	Partially Observable Markov Decision Process

References

1. Moradi, S.; Bokani, A.; Hassan, J. UAV-based Smart Agriculture: A Review of UAV Sensing and Applications. In Proceedings of the 2022 32nd International Telecommunication Networks and Applications Conference (ITNAC), Wellington, New Zealand, 30 November–2 December 2022.
2. Velusamy, P.; Rajendran, S.; Mahendran, R.K.; Naseer, S.; Shafiq, M.; Choi, J.G. Unmanned Aerial Vehicles (UAV) in precision agriculture: Applications and challenges. *Energies* **2021**, *15*, 217. [CrossRef]
3. Zhang, S.; Liu, J. Analysis and optimization of multiple unmanned aerial vehicle-assisted communications in post-disaster areas. *IEEE Trans. Veh. Technol.* **2018**, *67*, 12049–12060. [CrossRef]
4. Ashour, R.; Aldhaheri, S.; Abu-Kheil, Y. Applications of UAVs in Search and Rescue. In *Unmanned Aerial Vehicles Applications: Challenges and Trends*; Springer: Cham, Switzerland, 2023.
5. Bondi, E.; Kapoor, A.; Dey, D.; Piavis, J.; Shah, S.; Hannaford, R.; Iyer, A.; Joppa, L.; Tambe, M. Near Real-Time Detection of Poachers from Drones in AirSim. In Proceedings of the IJCAI, Stockholm, Sweden, 13–19 July 2018.
6. Zheng, W.; Wang, H.; Ji, H.; Wu, J. UAV formation flight and collision warning with centralized control of ground control station. In Proceedings of the 2019 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, 17–19 October 2019.
7. Wang, J.; Jiao, Z.; Chen, J.; Hou, X.; Yang, T.; Lan, D. Blockchain-Aided Secure Access Control for UAV Computing Networks. *IEEE Trans. Netw. Sci. Eng.* **2023**, *11*, 5267–5279. [CrossRef]
8. Li, H.; Cai, Y.; Hong, J.; Xu, P.; Cheng, H.; Zhu, X.; Hu, B.; Hao, Z.; Fan, Z. VG-Swarm: A Vision-Based Gene Regulation Network for UAVs Swarm Behavior Emergence. *IEEE Robot. Autom. Lett.* **2023**, *8*, 1175–1182. [CrossRef]
9. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef]

10. Wang, C.; Wang, J.; Shen, Y.; Zhang, X. Autonomous navigation of UAVs in large-scale complex environments: A deep reinforcement learning approach. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2124–2136. [CrossRef]
11. Hu, K.; Li, H.; Zhuang, J.; Hao, Z.; Fan, Z. Efficient Focus Autoencoders for Fast Autonomous Flight in Intricate Wild Scenarios. *Drones* **2023**, *7*, 609. [CrossRef]
12. Huang, H.; Zhu, G.; Fan, Z.; Zhai, H.; Cai, Y.; Shi, Z.; Dong, Z.; Hao, Z. Vision-based Distributed Multi-UAV Collision Avoidance via Deep Reinforcement Learning for Navigation. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022.
13. Cubuk, E.D.; Zoph, B.; Mané, D.; Vasudevan, V.; Le, Q.V. AutoAugment: Learning Augmentation Strategies from Data. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 113–123. [CrossRef]
14. Raileanu, R.; Goldstein, M.; Yarats, D.; Kostrikov, I.; Fergus, R. Automatic data augmentation for generalization in reinforcement learning. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 5402–5415.
15. Hansen, N.; Wang, X. Generalization in reinforcement learning by soft data augmentation. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 13611–13617.
16. Cheng, Y.; Zhao, P.; Wang, F.; Block, D.J.; Hovakimyan, N. Improving the Robustness of Reinforcement Learning Policies with ℓ_1 Adaptive Control. *IEEE Robot. Autom. Lett.* **2022**, *7*, 6574–6581. [CrossRef]
17. Shi, G.; Zhang, J.; Li, H.; Wang, C. Enhance the performance of deep neural networks via L2 regularization on the input of activations. *Neural Process. Lett.* **2019**, *50*, 57–75. [CrossRef]
18. Wang, K.; Kang, B.; Shao, J.; Feng, J. Improving Generalization in Reinforcement Learning with Mixture Regularization. In *Proceedings of the Advances in Neural Information Processing Systems*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H., Eds.; Curran Associates, Inc.: San Francisco, CA, USA, 2020; Volume 33, pp. 7968–7978.
19. Li, C.; Wang, T.; Wu, C.; Zhao, Q.; Yang, J.; Zhang, C. Celebrating diversity in shared multi-agent reinforcement learning. In Proceedings of the 35th International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 6–14 December 2024; NIPS '21.
20. Ghosh, P.; Sajjadi, M.S.; Vergari, A.; Black, M.; Scholkopf, B. From Variational to Deterministic Autoencoders. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
21. Zhang, A.; Lyle, C.; Sodhani, S.; Filos, A.; Kwiatkowska, M.; Pineau, J.; Gal, Y.; Precup, D. Invariant causal prediction for block mdps. In Proceedings of the International Conference on Machine Learning, Virtual, 13–18 July 2020.
22. Schölkopf, B.; Locatello, F.; Bauer, S.; Ke, N.R.; Kalchbrenner, N.; Goyal, A.; Bengio, Y. Toward causal representation learning. *Proc. IEEE* **2021**, *109*, 612–634. [CrossRef]
23. Song, X.; Jiang, Y.; Tu, S.; Du, Y.; Neyshabur, B. Observational Overfitting in Reinforcement Learning. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
24. Zhang, A.; McAllister, R.T.; Calandra, R.; Gal, Y.; Levine, S. Learning Invariant Representations for Reinforcement Learning without Reconstruction. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 30 April 2020.
25. Li, Z.; Cai, R.; Chen, G.; Sun, B.; Hao, Z.; Zhang, K. Subspace Identification for Multi-Source Domain Adaptation. *Adv. Neural Inf. Process. Syst.* **2024**, *36*, 1–15.
26. Goss, J.; Rajvanshi, R.; Subbarao, K. Aircraft conflict detection and resolution using mixed geometric and collision cone approaches. In Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Providence, RI, USA, 16–19 August 2004; p. 4879.
27. Seo, J.; Kim, Y.; Kim, S.; Tsourdos, A. Collision avoidance strategies for unmanned aerial vehicles in formation flight. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 2718–2734. [CrossRef]
28. Lin, Z.; Castano, L.; Mortimer, E.; Xu, H. Fast 3D collision avoidance algorithm for fixed wing UAS. *J. Intell. Robot. Syst.* **2020**, *97*, 577–604. [CrossRef]
29. Pérez-Carabaza, S.; Scherer, J.; Rinner, B.; López-Orozco, J.A.; Besada-Portas, E. UAV trajectory optimization for Minimum Time Search with communication constraints and collision avoidance. *Eng. Appl. Artif. Intell.* **2019**, *85*, 357–371. [CrossRef]
30. Biswas, S.; Anavatti, S.G.; Garratt, M.A. A particle swarm optimization based path planning method for autonomous systems in unknown terrain. In Proceedings of the 2019 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), Piscataway, NJ, USA, 1–3 July 2019; pp. 57–63.
31. Zsedrovits, T.; Zarándy, Á.; Vanek, B.; Péni, T.; Bokor, J.; Roska, T. Collision avoidance for UAV using visual detection. In Proceedings of the 2011 IEEE International Symposium of Circuits and Systems (ISCAS), Rio de Janeiro, Brazil, 15–18 May 2011; pp. 2173–2176.
32. Liang, Q.; Wang, Z.; Yin, Y.; Xiong, W.; Zhang, J.; Yang, Z. Autonomous aerial obstacle avoidance using LiDAR sensor fusion. *PLoS ONE* **2023**, *18*, e0287177. [CrossRef]

33. Qie, H.; Shi, D.; Shen, T.; Xu, X.; Li, Y.; Wang, L. Joint optimization of multi-UAV target assignment and path planning based on multi-agent reinforcement learning. *IEEE Access* **2019**, *7*, 146264–146272. [CrossRef]
34. Xue, Y.; Chen, W. Multi-agent deep reinforcement learning for uavs navigation in unknown complex environment. *IEEE Trans. Intell. Veh.* **2023**, *9*, 2290–2303. [CrossRef]
35. Fei, W.; Xiaoping, Z.; Zhou, Z.; Yang, T. Deep-reinforcement-learning-based UAV autonomous navigation and collision avoidance in unknown environments. *Chin. J. Aeronaut.* **2024**, *37*, 237–257.
36. Xie, S.; Zheng, Z.; Chen, L.; Chen, C. Learning semantic representations for unsupervised domain adaptation. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018.
37. Liu, A.H.; Liu, Y.C.; Yeh, Y.Y.; Wang, Y.C.F. A unified feature disentangler for multi-domain image translation and manipulation. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 1–10.
38. Peng, X.; Huang, Z.; Sun, X.; Saenko, K. Domain agnostic learning with disentangled representations. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019.
39. Cai, R.; Li, Z.; Wei, P.; Qiao, J.; Zhang, K.; Hao, Z. Learning disentangled semantic representation for domain adaptation. In Proceedings of the IJCAI, Macao, China, 10–16 August 2019.
40. Kong, L.; Xie, S.; Yao, W.; Zheng, Y.; Chen, G.; Stojanov, P.; Akinwande, V.; Zhang, K. Partial disentanglement for domain adaptation. In Proceedings of the International Conference on Machine Learning, Baltimore, MD, USA, 17–23 July 2022.
41. Shah, S.; Dey, D.; Lovett, C.; Kapoor, A. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In Proceedings of the Field and Service Robotics: Results of the 11th International Conference, Zurich, Switzerland, 12–15 September 2017; Springer: Cham, Switzerland, 2018; pp. 621–635.
42. Lu, C.; Wu, Y.; Hernández-Lobato, J.M.; Schölkopf, B. Invariant causal representation learning for out-of-distribution generalization. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4 May 2021.
43. Khemakhem, I.; Kingma, D.; Monti, R.; Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework. In Proceedings of the International Conference on Artificial Intelligence and Statistics, San Diego, CA, USA, 26–28 August 2020; pp. 2207–2217.
44. Tian, J.; Pearl, J. *Causal Discovery from Changes: A Bayesian Approach, Ucla Cognitive Systems Laboratory*; Technical Report; UCLA Computer Science: Los Angeles, CA, USA, 2001.
45. Long, M.; Zhu, H.; Wang, J.; Jordan, M.I. Deep transfer learning with joint adaptation networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 11 August 2017.
46. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2014**, arXiv:1312.6114.
47. Pearl, J. *Causality: Models, Reasoning and Inference*; Cambridge University Press: Cambridge, UK, 2009.
48. Tziavou, O.; Pytharouli, S.; Souter, J. Unmanned Aerial Vehicle (UAV) based mapping in engineering geological surveys: Considerations for optimum results. *Eng. Geol.* **2018**, *232*, 12–21. [CrossRef]
49. Hart, P.E.; Nilsson, N.J.; Raphael, B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. Syst. Sci. Cybern.* **1968**, *4*, 100–107. [CrossRef]
50. Dijkstra, E.W. A note on two problems in connexion with graphs. In *Edsger Wybe Dijkstra: His Life, Work, and Legacy*; ACM Books: New York, NY, USA, 2022; pp. 287–290.
51. Ma, J.; Yang, L.; Sun, Q. Capped L1-norm distance metric-based fast robust twin bounded support vector machine. *Neurocomputing* **2020**, *412*, 295–311. [CrossRef]
52. Singla, S.; Singla, S.; Feizi, S. Improved deterministic l2 robustness on CIFAR-10 and CIFAR-100. *arXiv* **2021**, arXiv:2108.04062.
53. Han, G.; Wu, Q.; Wang, B.; Lin, C.; Zhuang, J.; Li, W.; Hao, Z.; Fan, Z. Deep Reinforcement Learning based Multi-UAV Collision Avoidance with Causal Representation Learning. In Proceedings of the 10th International Conference on Big Data and Information Analytics (BigDIA), Chiang Mai, Thailand, 25–28 October 2024.
54. Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.P.; Glorot, X.; Botvinick, M.M.; Mohamed, S.; Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR (Poster)* **2017**, *3*, 1–13.
55. Kim, H.; Mnih, A. Disentangling by factorising. In Proceedings of the International Conference on Machine Learning. PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 2649–2658.
56. Estermann, B.; Wattenhofer, R. DAVA: Disentangling Adversarial Variational Autoencoder. *arXiv* **2023**, arXiv:2303.01384.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.