CAUSAL EXPLANATIONS FOR HUMAN UNDERSTANDING IN DEEP NEURAL POLICIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Explainable deep learning models are important for the development, certification, and adoption of autonomous systems. Yet, without methods to quantify causal relationships between explanations and actions, interpretability remains correlational. Furthermore, explanations typically address lower-level actions. This poorly serves human understanding, which benefits from higher-level abstractions, and underactuated robotics, whose behaviors often require richer descriptions. To address these gaps, we introduce Causal Concept-Wrapper Network (CCW-Net), a general training method across differentiable architectures that adapts mediation analysis from fields such as economics, medicine, and epidemiology to align the causal effects of abstract, information-rich explanations with policy actions. CCW-Net expands the expressiveness of prior work in both explainable deep learning and mediation analysis allowing each explanation to serve as a mediator encoding both its presence and context-based expression. In a high-fidelity, underactuated aircraft formation task, CCW-Net produces high-level explanations that are both interpretable and quantifiably causal without degrading task performance. We demonstrate CCW-Net across diverse architectures including capsule networks with dynamic routing, modified concept bottleneck models, and cross-attention mechanisms. Notably, we present the first adaptation of capsule networks to sequential decision-making in robotics. This breadth shows that CCW-Net applies broadly across neural network architectures, offering a general path toward transparent and trustworthy autonomy.

1 INTRODUCTION

Learned policies in autonomous systems are moving from research labs into high-stakes, real-world settings such as aviation (Pope et al., 2022; Ward, 2023), driving (Phan-Minh et al., 2023), and robotics (Tang et al., 2025). In task and safety-critical domains such as these, interpretability is becoming a necessary component (Rudin, 2019; Atakishiyev et al., 2025). Developers need it to troubleshoot (Kenny et al., 2024), testers need it to verify (Rountree et al., 2021; Mahmud et al., 2024a), and users need it to understand (Sanneman & Shah, 2022). Yet many neural policies remain black boxes: they are not interpretable in a way that is useful for people in the task at hand.

Recent concept-based interpretability offers promise by structuring decisions through human-interpretable concepts (Koh et al., 2020; Echterhoff et al., 2024). However, prior work typically represents concepts as scalars (Madumal et al., 2020; Koh et al., 2020; Kenny et al., 2024), for example, a "left turn" in driving or the "build supply depot" action in Starcraft II. Scalars are useful for indicating the presence or strength of a concept, but they cannot capture the higher-dimensional structure needed to communicate complex maneuvers or concepts that can manifest themselves in many ways. Additionally, increased abstraction allows people to convey information in denser, richer units that balance user workload and understanding (Sanneman et al., 2024). In this work, we extend concept representation from scalars to vectors, enabling each concept to encode both *whether* it exists and *how* it is expressed bringing concept-based explanations closer to the richness of human reasoning (Tucker et al., 2022).

Moreover, the relationship between concepts and actions remains largely correlational, i.e., a policy might activate a concept without that concept necessarily causing the observed action, leading to spurious explanations (Zhou et al., 2022). A vehicle policy might activate a "passenger pickup"

concept but it does not mean that such concept caused the observed action. The model may appear to explain its actions when in reality relying on spurious correlations and confounding patterns in the data. To be useful in troubleshooting, certifying, and understanding explanations must capture causal effects or what would happen if we intervened to change a concept while holding others fixed (Pearl, 2012). Correlation, no matter how sophisticated, does not address causal questions (Pearl, 2009). Misleading explanations can obscure risks, prevent effective troubleshooting, and erode user trust (Wang et al., 2022). For safety and task-critical autonomy we must transition from correlation to causation.

We address these challenges with Causal Concept Wrapper Network (CCW-Net), a training method that adapts causal mediation analysis, well established in fields such as economics and epidemiology (Celli, 2022; Lee et al., 2021), to the design of interpretable policies. CCW-Net estimates the causal effect of each concept on actions and trains a policy to align its concept representations with these effects. In doing so, it produces explanations that are both human-meaningful and quantifiably causal while expanding the representational capacity of concepts beyond scalars to high-dimensions.

Main contributions Our contributions are fourfold:

1. **Causal Concept Attribution:** A general method for computing and training per-concept action attributions that target *causal* effects, not just correlations.
2. **Causal Alignment through Mediation Analysis:** Adoption of interventional mediation analysis to estimate each concept's effect on actions and align the policy attributions accordingly.
3. **Concepts as Vectors:** Extend concept bottleneck models to represent concepts as vectors capturing contextual structure enabling abstract, human-aligned explanations.
4. **Architecture Agnostic:** A general framework broadly applicable to any differentiable policy head, independent of architecture choice and task domain, enabling broad integration.

2 Related Work

We now briefly discuss a few related research directions. Additional discussion is in Appendix A.

Concept-based Explanations The idea of using concepts to generate explanations of AI systems is widely explored (Kim et al., 2018; Alvarez-Melis & Jaakkola, 2018; Koh et al., 2020; Bai et al., 2023; Achtibat et al., 2023; Tan et al., 2024). These methods have found applications in variety of fields, including biomedical applications (Graziani et al., 2018; Clough et al., 2019; Yeche et al., 2019), scientific research (Sprague et al., 2019; Yang et al., 2024), game-playing systems (Lovering et al., 2022; Tomlin et al., 2022; Schut et al., 2025), planning agents (Kazhdan et al., 2021; Qian et al., 2024), etc. These concepts can come from various sources: as input from human experts (Ghandeharioun et al., 2022), or by extracting them from labeled data (Ghorbani et al., 2019; Yeh et al., 2020). In this work, we use a mix of both the approaches, where a domain expert provides the concepts as input and using those we created auto-labeler for given data.

Concept-Based Interpretable Policies Recent work has extended concept-based explanations to sequential decision making (Koh et al., 2020; Kenny et al., 2023; Das et al., 2023). Concept Bottleneck Models (Koh et al., 2020) first showed that a neural network could be trained to predict concepts then use them to make downstream decisions over those concepts. More recent approaches in imitation learning, such as PW-Net (Kenny et al., 2023) and CW-Net (Kenny et al., 2024), adapt this idea to control tasks, showing that concept-based explanations can support user understanding and trust calibration. However, these methods typically represent each concept as a single scalar, and do not establish causal relationships between concepts and actions. While scalars can indicate whether and degree to which a concept is present, they lack expressiveness to capture higher-level abstractions such as the *manner* of lane change for a given context. Also, they need not causally establish the relationship between the high-level concept and the exact action to be executed.

Causal Mediation Analysis In parallel, causal inference methods have developed sophisticated tools for establishing causal relationships from observational data. Other fields such as epidemiology, economics, and medicine have developed mature tools for causal mediation analysis (MacKinnon, 2008; Imai et al., 2010; VanderWeele, 2015). Causal mediation analysis decomposes treatment

effects into pathways through mediators (Pearl, 2009; Loh et al., 2022), with the interventional indirect effect (IIE) quantifying how much of a treatment's effect operates through specific mediators. These methods are identifiable from observational data under standard ignorability conditions but have not been adapted to establish causal relationships between learned concept representations and model outputs in deep learning. In this direction, our contribution is to adapt these causal tools to imitation learning and extend their treatment of individual mediators from scalars to vectors.

Causal Attribution Methods in Deep Learning Finally, a wide range of attribution methods methods aim to explain deep networks by linking inputs or intermediate features to outputs. Examples include saliency maps (Simonyan & Zisserman, 2015; Selvaraju et al., 2017), perturbation methods (Ribeiro et al., 2016; Lundberg & Lee, 2017), and gradient based techniques (Smilkov et al., 2017; Sundararajan et al., 2017). While these methods can highlight what features are associated with a decision, they generally remain correlational. For instance, ablating a feature may change a prediction, but this does not establish that the feature causally drives the outcome (Adebayo et al., 2018; Zhou et al., 2022). In safety and task-critical autonomy, such correlational explanations can be misleading. Our approach differs by explicitly grounding attribution in estimated causal effects. We not only compute how actions respond to concept changes, but also check these sensitivities against interventional baselines drawn from data.

3 Problem Formulation

As mentioned earlier, in this work, we address the challenge of training interpretable neural policies that provide causally grounded explanations for their actions. Given a set of trajectories and expert human-defined concepts, we seek to learn a policy that: (1) matches expert performance, (2) reasons through interpretable concepts, and (3) ensures that concept-action relationships reflect true causal effects rather than spurious correlations.

Input: We assume access to: (1) a pretrained black-box policy f that achieves good task performance but lacks interpretability, and (2) expert trajectories $\mathcal{T} = \{(X_i, Y_i, c_i^\ell)\}_{i=1}^N$ where $X_i \in \mathcal{X}$ are observation states (e.g., sensor readings, game states), $Y_i \in \mathcal{Y}$ are expert actions, and c_i^ℓ are concept labels indicating which human-interpretable concepts are active (e.g., "lead", "lag").

Desired Output: An interpretable policy $\pi_\theta : \mathcal{X} \to \hat{\mathcal{Y}}$ that achieves expert-level task performance, provides concept-based explanations for each output action $\hat{Y} \in \hat{\mathcal{Y}}$, and ensures explanations reflect causal relationships between concepts and actions. We must also jointly optimize imitation, concept classification, and causal alignment. More details about its mathematical formulation are in Sec. 5.3.

Assumptions: We assume the following in our setup: (i) the provided human concepts capture the key decision-making factors for the task; (ii) all causal pathways from observations to actions can be mediated through the concept representations; (iii) the standard ignorability conditions hold (discussed in Section 5.1); and the expert demonstrations provide reliable ground truth for both task performance and concept labeling.

4 Preliminaries

Structural Causal Models A structural causal model (SCM) (Pearl, 2009; Peters et al., 2017) consists of a set of equations, $X_i = k_i(pa_i, u_i), i = 1, \ldots, m$, where each equation represents an autonomous mechanism that determines the value of exactly one distinct variable; X_i and u_i are the i-th random variable and its corresponding error term, respectively. The function k_i represents the causal mechanism generating X_i, and pa_i denotes the set of variables that directly cause X_i, i.e., are parent variables of X_i.

Mediation Analysis Mediation analysis (Pearl, 2012) decomposes the causal effect of a treatment A on an outcome Y into pathways that operate through intermediate variables called mediators M. For a treatment A, outcome Y, and mediators $M = \langle c_1, \ldots . c_J \rangle$, the total effect *TE* decomposes as $TE = DE + IE$, where *DE* are the direct effects, and *IE* are the indirect effects. For multiple concepts,

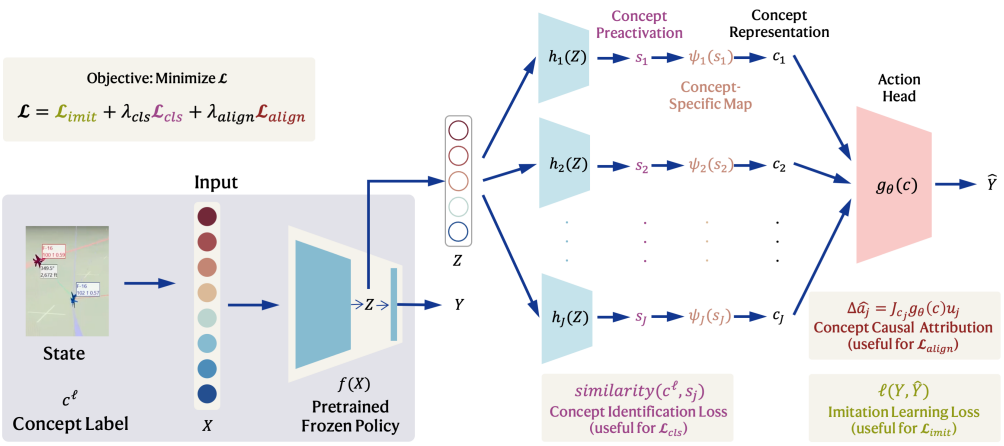


Figure 1: **CCW-Net Architecture**. CCW-Net wraps a frozen pre-trained policy with a concept module $h(z)$ and policy head $g_\theta(c)$ to produce causally grounded explanations. The method estimates causal targets IE_j from expert data, computes local concept-action effects $\Delta\hat{a}_j$ via Jacobians, and aligns them using a cosine loss. Vector concepts encode both presence and context-dependent expression, enabling rich interpretable representations while preserving task performance.

we can further decompose indirect effects as $IE = \sum_j IE_j$ for all concepts j. Furthermore, following Loh et al. (2022), we relax assumptions on the causal graph by introducing an interaction term, IE_μ. Therefore the causal decomposition of total effects is $TE = \sum_j IE_j + IE_\mu$.

Imitation Learning Imitation learning trains a parametric policy π_θ to replicate expert behavior from demonstrated state-action pairs $\{(X_i, Y_i)\}$ from the trajectories $\mathcal{T}$ (Zare et al., 2024). The policy parameters θ are optimized by minimizing the expected imitation loss $\min_\theta \frac{1}{N}\sum_{i=1}^{N} \ell(\pi_\theta(X_i), Y_i)$, where $\ell(\cdot,\cdot)$ measures the discrepancy between predicted and expert actions. This supervised learning approach assumes the expert demonstrations are optimal or near-optimal for the task, enabling the learned policy to reproduce expert performance without requiring explicit reward engineering.

5 Causal Concept Wrapper Network (CCW-Net)

We now describe our approach to solve the problem described in Sec. 3. We develop a Causal Concept Wrapper Network (CCW-Net), which enhances a pretrained black-box, deep neural policy with human-interpretable explanations that are quantifiably causal to the policy's actions. It wraps the input frozen backbone policy with CCW-Net's concept module h_θ and policy head g_θ and is trained to (i) imitate the expert, (ii) predict human-interpretable concept labels, and (iii) align each concept's local effect on the action with a causal target estimated from expert trajectories.

Architecture Kenny et al. (2023; 2024) show that concept wrapper architectures can enhance pretrained black-box policies with human-interpretable explanations by introducing an intermediate concept representation layer. We leverage this to develop CCW-Net as shown in Figure 1. Given an input frozen non-interpretable policy network $f : \mathcal{X} \to \mathcal{Y}$ that maps observation state $X \in \mathcal{X}$ to an action $Y \in \mathcal{Y}$, we extract the latent vector $Z \in \mathcal{Z}$ from f, which are used as input to the last layer in f. Note that this last layer in policy network f, takes Z as input to produce Y as output. We then use a concept module $h_\theta(Z)$ that produces human-interpretable vector concepts s_j and their activations c_j, and add a new policy head $g_\theta(c)$.

Causal roles and notation We model CCW-Net's wrapper as SCM $\mathcal{G}_{\text{CCW-Net}} = \{\mathcal{X}, \mathcal{Z}, M, \hat{\mathcal{Y}}\}$ with observations $\mathcal{X}$, black-box policy latent $\mathcal{Z}$, concept mediators M, and actions $\hat{\mathcal{Y}}$. We represent each logged expert trajectory sample with concept labels a tuple $\langle C, A, M, Y\rangle$, consisting of covari-

ants C used to represent the raw observed features $X \in \mathcal{X}$[1]; treatments A which correspond to a binary concept activation, representing if a concept is active or not; mediators $M = \langle c_1, \ldots, c_J \rangle$ used to represent vectors c_j corresponding to each concept $j \in 1, \ldots, J$; and outcomes $\hat{\mathcal{Y}}$ representing the expert action generated by interpretable policy π_θ. CCW-Net models each concept c_i as a mediator between the frozen backbone's latent representation Z and the policy head $g_\theta(c)$. Beyond standard imitation, CCW-Net adds a causal-alignment objective that encourages the model's local, per-concept action effect to match the interventional indirect effect (IIE) estimated from trajectories.

We jointly train the concept module and action head to imitate the expert while aligning each concept's local effect with its IIE target. We use a directional Jacobian per concept block to measure the causal attribution, hence CCW-Net is policy head agnostic so long as pathways from concepts to actions are differentiable.

Algorithm 1: CCW-Net (Causal Concept-Wrapper Network)

Input: Data $\mathcal{D} = \{(C_i, A_i, Y_i)\}$; frozen backbone; modules h_θ, g_θ; $\lambda_{\text{align}}, \lambda_\mu, S, \delta, \tau$

1 **Initialize:** Build treatment group sets $\mathcal{B}$ with $(C_i, M_i = h_\theta(z_i))$ tuples and normalization stats; save frozen head parameters (cross-fitted);
2 **for** *epoch* $= 1, \ldots, E$ **do**
3 **if** *refresh time* **then**
4 Partially update sets
5 (e.g., 10% replacement)
6 **for** *batch* $(C, A, Y) \sim \mathcal{D}$ **do**
7 **foreach** *sample* i **do**
8 Draw S kernel-matched mediators from $\mathcal{B}_i$ (caliper δ, temp τ);
9 Compute IE_j, TE, IE_μ via concept exchanges with frozen head;
10 $c \leftarrow h_\theta(\text{backbone}(C))$;
11 $\hat{Y} \leftarrow g_\theta(c)$;
12 $\Delta\hat{a}_j \leftarrow J_{c_j} g_\theta(c) u_j$ for active j;
13 $\mathcal{L} \leftarrow \mathcal{L}_{\text{imit}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{causal}}\mathcal{L}_{\text{causal}}$;
14 Update θ via backprop;

Algorithm We divide the CCW-Net framework's operation (Alg. 1) in three main phases:

Phase 1: Estimate causal targets from data. Partition data from expert trajectory logs by treatment group A, build mediator reference sets per group, draw kernel-matched mediators within group, and perform within-sample concept swaps to estimate IE_j, TE, and the residual, IE_μ.

Phase 2: Compute policy's concept-action effects. For each concept j compute a directional local effect $\Delta\hat{a}_j = J_{c_j} g_\theta(c) u_j$, where J_{c_j} is the Jacobian for concept j in unit direction $u_j = c_j / \|c_j\| + \varepsilon$, to determine how the action would change for a given change in concept representation.

Phase 3: Align data and policy effects. Align each concept j's local effect with its estimated IE_j by minimizing $\sum_j m_j(1 - \cos(\Delta\hat{a}_j, IE_j))$ over active concept mask $m_j \in \{0, 1\}$ together with the imitation loss.

5.1 Phase 1: Estimate Causal Training Targets from Data

As mentioned earlier, we represent each logged expert trajectory sample with concept labels as a tuple $\langle C, A, M, Y \rangle$. This is permissible because we adopt interventional mediation analysis with the interventional indirect effect (IIE) from Vansteelandt & Daniel (2017) as the per-concept causal target. Under standard ignorability conditions (no unmeasured confounding of $A \to Y$ given C, of $M \to Y$ given A, C, and of $A \to M$ given C), IIEs are identifiable from observational data *without* specifying a causal ordering among mediators (Loh et al., 2022). This is important because there may be context-dependent interactions between concepts for a given task or wrapper architecture.

Identification. We adopt interventional mediation analysis (Vansteelandt & VanderWeele, 2012) extended to multiple mediators (Vansteelandt & Daniel, 2017) and the interventional indirect effect (IIE) as our causal target. Under standard ignorability conditions (no unmeasured confounding of $A \to Y$ given C, of $M \to Y$ given A, C, and of $A \to M$ given C), IIEs are identifiable from observational data *without* specifying a causal ordering among mediators (Loh et al., 2022).

Treatment group reference sets. We partition data in groups by the treatment A (i.e., concept activations), and for each group maintain a reference set of tuples $\mathcal{B} = (C_i, M_i)$ (line 1 in Alg. 1). Given a query sample with covariates C^* and treatment A^*, we approximate the interventional

[1] C can also represent frozen latent features $Z \in \mathcal{Z}$, if trajectories available in form of latent features

mediator distribution $P(M \mid A = b,\ C \approx C^*)$ via kernel matching within that treatment group: (i) standardize features within the group, (ii) compute scaled squared distances covariate space $D_i = \|\tilde{C} - \tilde{C}_i\|_2^2/p$, (iii) discard neighbors with $D_i > \delta^2$ (caliper δ), (iv) convert similarities to sampling weights: $w_i \propto \exp(-D_i/2\tau^2)$, such that $\sum_i w_i = 1$. If no admissible neighbors remain, we fall back to the nearest valid neighbor. During training, we monitor support through effective sample size, coverage, and fallback rate. Additional details about it are in Appendix D.3.3.

Monte Carlo counterfactual estimation. As shown in lines 7-9 in Alg. 1, for each sample, we draw S matched mediator sets $M^{(s)}$ from the corresponding treatment group reference set and form counterfactual mediators by exchanging one active concept (holding other concepts fixed). Exchanging all active concepts yields the joint intervention. We obtain counterfactual actions $Y_\theta(\cdot)$ by passing these mediator sets through the frozen policy head parameters stored with the reference set (cross-fitted to reduce bias). Averaging over S draws gives the following Monte-Carlo estimates:

(i) **Total effect:** $TE = Y_{c_{j,obs}} - Y_{c_{j,ref}}$, the action change from exchanging both active concepts; (ii) **Per-Concept IIE:** $IE_j = Y_{c_{j,obs}} - Y_{c_{j,ref}, c_{-j,obs}}$, the action change from exchanging only concept j, with others fixed; and (iii) **Residual Interaction:** $IE_\mu = TE - \sum_j IE_j$, the mediator interactions.

For cases where *mutually exclusive concepts* exist, define individual exchanges as deactivating the concept of interest and activating its mutually exclusive pair. For the joint exchange, swap the active slots on all mutually exclusive pairs. Further discussion is in Appendix E.2.1.

No direct effect by construction. By construction, we only intervene through mediators since there is no direct $\mathcal{Z} \to A$ pathway in CCW-Net's head. By definition, the direct effect is zero.

Mediator interactions. For analysis, Appendix D.3.2 reports the directional interaction share, $\phi_\mu = \langle IE_\mu, TE\rangle / \|TE\|^2$, which is signed and can be negative when interactions oppose *TE*.

5.2 Phase 2: Compute Policy's Effects of Concepts on Actions

Wrapper and notation. As in lines 10-11, concept module h_θ consumes the frozen backbone latent Z and produces per-concept pre-activations $s_j \in \mathbb{R}^{d_j}$ with per-concept normalization ψ_j that produces concept representations $c_j = \psi_j(s_j)$. The policy head g_θ takes $c = (c_1, \ldots, c_J)$ and produces predicted actions $\hat{Y} = g_\theta(c)$, including final nonlinearities (e.g., $tanh$).

Per-concept causal attribution. For each concept j, we quantify the policy's instantaneous action sensitivity to that concept by defining a radial unit vector in pre-activation concept space, $u_j^s = s_j/(\|s_j\| + \varepsilon)$, and compute a Jacobian through the composed map $g_\theta \circ \psi$, $\Delta\hat{a}_j = J_{s_j}(g_\theta \circ \psi)(s), u_j^s \in \mathbb{R}^A$. Equivalently, in concept space this can be written as $\Delta\hat{a}_j = J_{c_j} g_\theta(c), u_j^c$, where the direction u_j^c is obtained by propagating u_j^s through the normalization mapping, $u_j^c = J_{s_j}\psi_j(s_j), u_j^s$ (line 12 in Alg. 1). This approach is head-agnostic so long as the concept-to-action mapping is differentiable. Expressing the local effect in $\Delta\hat{a}_j$ with ψ ensures that the causal alignment loss is well-defined across differentiable architectures.

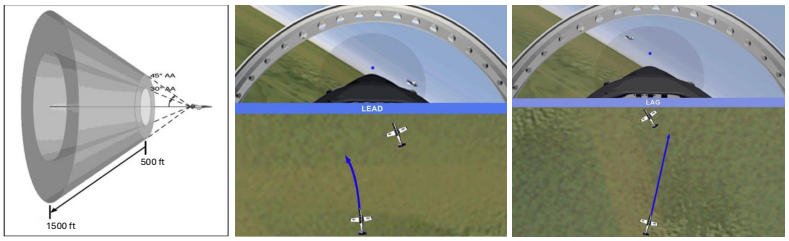
Concept supervision. In this work, concepts are supervised on s_j. This corresponds to calculating $\mathcal{L}_{cls}$ by checking similarity of the input concept label(s) c^ℓ for the current state X with the predicted s_j. Based on the architecture and the task, it can be implemented using any compatible method. For e.g., we use softmax cross-entropy for our CBM and attention head, and a margin loss (Sabour et al., 2017) for our capsule network head.

5.3 Phase 3: Align Policy Effects to Causal Targets

Directional alignment. For each sample we align the local effect $\Delta\hat{a}_j$ to the causal target IE_j with a masked cosine objective:

$$\mathcal{L}_{\text{align}}(x) = \sum_j m_j(x)(1 - \cos(\Delta\hat{a}_j, IE_j(x))), \quad \cos(u, v) = \frac{\langle u, v\rangle}{\|u\|\|v\| + \varepsilon}.$$

where $m_j(x) \in 0, 1$ masks inactive concepts.



(a) (b) (c)

Figure 2: Extended trail task. (a) Extended trail cone. The chase aircraft (not depicted) is tasked with remaining within the dark shaded region of this cone while the lead aircraft (depicted) maneuvers (United States Air Force, 2024). (b) Lead pursuit. Performed by pointing the chase aircraft's nose ahead of the lead aircraft. Top: View from the chase aircraft cockpit. Bottom: Top-down view. (c) Lag pursuit. Performed by pointing the chase aircraft's nose behind the lead aircraft. Top: View from the chase aircraft cockpit. Bottom: Top-down view (United States Air Force, 2025).

Full loss objective. As in line 13 of Alg. 1, we jointly optimize imitation, concept classification, and causal alignment as $\mathcal{L} = \mathcal{L}_{\text{imit}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}}$, where warmup epochs set $\lambda_{\text{align}} = 0$ with $\mathcal{L}_{\text{imit}}$ and $\mathcal{L}_{\text{cls}}$ active to allow the wrapper to build a representation of the task and concept before anchoring the model's causal effects from concepts to actions estimated from observational data and ensuring interpretability is not simply correlational, but causally valid.

Reference set bias. To limit estimation bias and leakage, the reference sets from Phase 1 are periodically refreshed (fully or partially) with cross-fitting during training. We monitor effective sample size, coverage, and fallback rates to ensure adequate support.

6 Empirical Evaluation

We evaluate CCW-Net in a real-world two-aircraft aircraft formation task, extended trail (United States Air Force, 2024), in a high-fidelity F-16 physics environment (So & Fan, 2023; Heidlauf et al., 2018). This setting stresses causal, abstract, human-interpretable explanations. Identical pitch and roll commands can serve different concepts depending on pursuit geometry and energy state so explanations in action space are ambiguous while abstract concepts are informative.

6.1 Aircraft Formation Task

Task. We evaluate CCW-Net in simulation on a real-world, complex, underactuated task: an aircraft formation task called extended trail United States Air Force (2024; 2025), shown in Figure 2, where a chase aircraft maintains position behind a lead aircraft while executing dynamic maneuvers. With it comes four concepts that are used by real-world pilots to describe, perform, and debrief the task: Lead pursuit, lag pursuit, climb, and dive that hereby form the concept set $M = \{\textit{Lead}, \textit{Lag}, \textit{Climb}, \textit{Dive}\}$. Additional details are discussed in Appendix C.

Environment. A control loop simulates aircraft dynamics in three dimensional with point-mass and six degree of freedom (Heidlauf et al., 2018). Two aircraft are simulated. The lead aircraft follows a scripted path while the chase aircraft was trained to maintain formation with reinforcement learning (So & Fan, 2023) (see Appendix C.2 for details). The state includes relative kinematics (e.g., relative range, angle, and closure rates), ownship state, and normalized energy parameters. The action space is two dimensional in pitch and roll command.

Expert policy and data. A reinforcement-learning expert generates demonstration trajectories. We log tuples $\langle X, Z, Y \rangle$ of raw observations X, frozen expert policy latents Z, and expert actions Y. Concept labels $A = (A_L, A_C) \in \{\textit{Lead}, \textit{Lag}\} \times \{\textit{Climb}, \textit{Dive}\}$ are obtained via human- or auto-labeling. Within each $\{\textit{Lead}, \textit{Lag}\}, \{\textit{Climb}, \textit{Dive}\}$ concepts are physically mutually exclusive and therefore only one concept is active (i.e., an aircraft cannot both *Climb* and *Dive*).

Why this task? Extended trail formation is a real-world, complex task in which pilots use the concepts $\{Lead, Lag, Climb, Dive\}$ to reason over, describe, and debrief with. These concepts are formally utilized in advanced flight training (United States Air Force, 2024; 2025) and serve to ground CCW-Net's utility for human-interpretability in complex real-world settings. From a robotics perspective, the task is not readily solved with inherently interpretable approaches. Furthermore, aircraft control, like many robotics applications, is underactuated making action-level labels poor proxies for human-meaningful concepts. The same action can implement different concepts depending on context. Additionally, sequences of actions are sometimes required to produce task-meaningful explanations.

6.2 Architectures Evaluated

Because $\Delta\hat{a}_j$ is a directional derivative, CCW-Net applies to any differentiable head mapping concepts to actions. We instantiate three heads to demonstrate architectural generality.

All approaches adapt a key insight: expanding concept bottleneck models' (Koh et al., 2020) representations of concepts from scalars to vectors. In this vector representation, we represent concept activations by vectors' lengths and concept expressions as their orientations. We then supervise concepts with labeled data. This enables policies to reason over context-dependent representations of arbitrarily abstract, human-interpretable concepts. Notably, where scalar-based concept representations were adequate for the driving policies in Kenny et al. (2023; 2024), they proved insufficient to reproduce the aircraft formation policy thereby motivating this work's extension to concept representations as vectors.

Vector CBM. We expand CBMs (Koh et al., 2020) to map $Z \to \{s_j\}_{j=1}^{4}$, producing vector logits per concept and form concept vectors v_j (softmax-normalized per block) and supervise two-way concept classifiers (Lead/Lag, Climb/Dive) with cross-entropy on block-level scores. The action head is a per-concept linear map summed across concepts, followed by $tanh$.

Capsule network with dynamic routing. We introduce capsule networks' (Hinton et al., 2011) first known use in predicting actions for sequential decision making systems. Concepts are capsule vectors $v_j = squash(s_j)$ whose lengths encode activation and orientations encode expression. Dynamic routing connects concept capsules to action capsules based on context. Concepts are supervised through a margin loss (Sabour et al., 2017). Actions are transformed via $tanh$. We compute $\Delta\hat{a}_j$ with Jacobians through the squash function, routing updates, and the final $tanh$.

Attention head. Each action dimension holds a learned query over concept key/values with learned K, V projections. We compute $\Delta\hat{a}_j$ via a Jacobian through a softmax attention. Across all heads, concepts are represented as vectors: length represents activation and orientation provides context-dependent expression enabling richer, human-interpretable abstractions.

7 Results

Hypotheses. We test four hypotheses: **H1:** CCW-Net increases concepts' causal alignment with policy actions across all tested architectures; **H2:** CCW-Net does not qualitatively degrade main task performance relative to the frozen backbone; **H3:** CCW-Net does not degrade main task performance relative to a baseline wrapper with λ_{align}=0; and **H4** CCW-Net does not degrade concept–classification accuracy relative to that baseline.

Results. CCW-Net substantially improves mean causal alignment for all three heads: CBM $+0.449$ ($p < 0.01$), Capsule $+0.368$ ($p < 0.001$), and Attention $+0.234$ (not significant). We interpret the causal alignment for the attention head to be meaningful given that it achieves the greatest causal alignment score (0.939 ± 0.026) (Figure 3 and Table 3) thereby supporting **H1**. Test MSE remains comparable to the baseline wrapper (CBM $+0.002$, Capsule $+0.003$, Attention $+0.001$); the Capsule delta is statistically significant ($p < 0.05$), but its magnitude (0.003) is operationally negligible and was confirmed by visualizing rollouts, supporting **H3**. Qualitatively, 2-minute evaluation rollouts showed no extended trail cone violations when compared against the frozen backbone, supporting **H2**. Concept classification was unchanged. Lead/Lag and Climb/Dive deltas are within

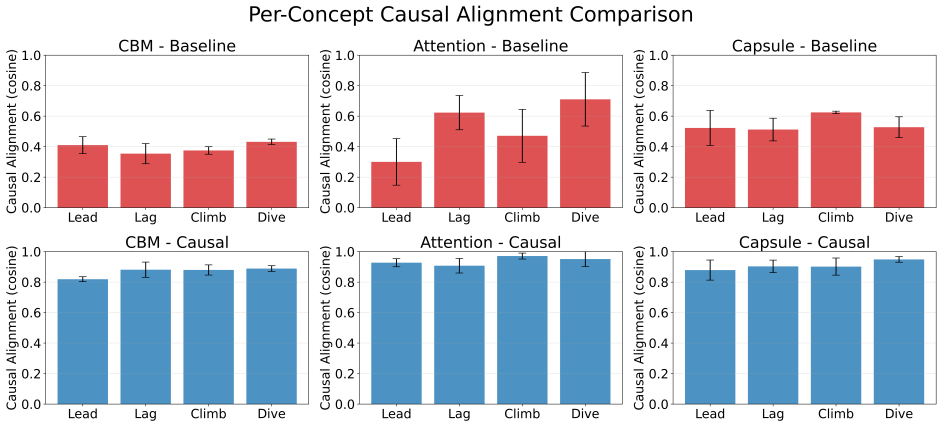


Figure 3: Causal Concept Alignment Before and After Causal Loss Applied. Red: CCW-Net Baseline with $\lambda_{\text{align}} = 0$. Blue: CCW-Net with $\lambda_{\text{align}} = 0.05$ applied. Causal alignment across all concepts and architectures is improved.

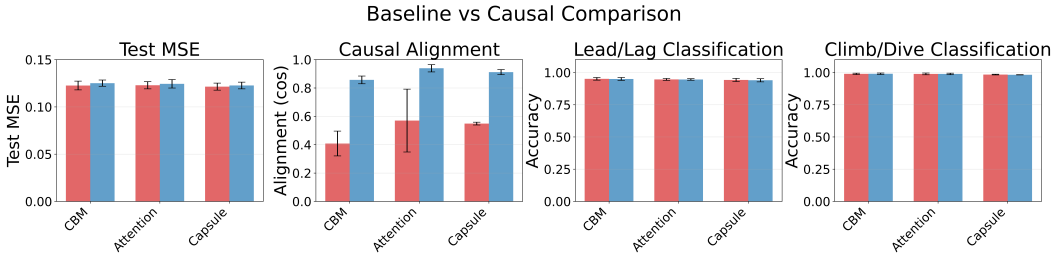


Figure 4: Change in Imitation MSE, Mean Causal Alignment, and Concept Classification Accuracy Due to Causal Losses. Red: CCW-Net Baseline with $\lambda_{\text{align}} = 0$. Blue: CCW-Net with $\lambda_{\text{align}} = 0.05$ applied. Causal alignment is improved while imitation and classification performance remain negligibly impacted.

0.0–0.3 percentage points and not significant across heads (Figure 4), supporting **H4**. Interestingly, training curves (Figure 5) suggest causal effects quickly take hold in the policy.

Summary. CCW-Net consistently increases concept-to-action causal alignment (**H1**) without degrading task performance vs. the frozen backbone (**H2**) or baseline wrapper (**H3**), and without harming concept accuracy (**H4**). The effect holds across Capsule, vector CBM, and Cross-Attention heads, indicating CCW-Net is a practical, architecture-agnostic path to causally grounded, human-friendly explanations.

8 Conclusion

The world is moving fast toward deploying increasingly capable autonomous systems empowered by deep neural policies into high-stakes, real-world settings, but without transparency into *why* actions are taken, real use in sensitive domains remains limited. We introduced CCW-Net, a flexible wrapper that enables any differentiable policy head to reason over human-defined concepts and aligning them with *causal* effects estimated from observed trajectories. On a high-fidelity flight task, CCW-Net consistently improved concept-to-action causal alignment across three architectures, concept bottleneck models, attention heads, and capsule networks, while maintaining high tasks performance and concept classification accuracy. CCW-Net offers an architecture-agnostic, causally grounded approach to interpretability of deep neural polices. In future work we look forward to extending causal connections to observations as well as producing causally-grounded counterfactual trajectories on the path toward enabling robots to answer ”why?”

REFERENCES

Reduan Achtibat, Maximilian Dreyer, Ilona Eisenbraun, Sebastian Bosse, Thomas Wiegand, Wojciech Samek, and Sebastian Lapuschkin. From attribution maps to human-understandable explanations through concept relevance propagation. *Nature Machine Intelligence*, 5(9):1006–1019, 2023.

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Proceedings of the 32nd Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

David Alvarez-Melis and Tommi S. Jaakkola. Towards robust interpretability with self-explaining neural networks. In *Proceedings of the 32nd Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Shahin Atakishiyev, Mohammad Salameh, and Randy Goebel. Safety implications of explainable artificial intelligence in end-to-end autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2025.

Andrew Bai, Chih-Kuan Yeh, Neil Y.C. Lin, Pradeep Kumar Ravikumar, and Cho-Jui Hsieh. Concept gradient: Concept-based interpretation without linear assumption. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*, 2023.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: Composable transformations of Python+NumPy programs, 2018. URL http://github.com/jax-ml/jax.

Viviana Celli. Causal mediation analysis in economics: Objectives, assumptions, models. *Journal of Economic Surveys*, 36(1):214–234, 2022.

James R. Clough, Ilkay Oksuz, Esther Puyol-Antón, Bram Ruijsink, Andrew P. King, and Julia A. Schnabel. Global and local interpretability for cardiac MRI classification. In *Proceedings of the 22nd International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2019.

Devleena Das, Sonia Chernova, and Been Kim. State2Explanation: Concept-based explanations to benefit agent learning and user understanding. In *Proceedings of the 37th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Giovanni De Felice, Arianna Casanova Flores, Francesco De Santis, Silvia Santini, Johannes Schneider, Pietro Barbiero, and Alberto Termine. Causally reliable concept bottleneck models. In *ICLR 2025 Workshop on XAI4Science: From Understanding Model Behavior to Discovering New Scientific Knowledge*, 2025.

Quentin Delfosse, Sebastian Sztwiertnia, Mark Rothermel, Wolfgang Stammer, and Kristian Kersting. Interpretable concept bottlenecks to align reinforcement learning agents. *Proceedings of the 38th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Gabriele Dominici, Pietro Barbiero, Mateo Espinosa Zarlenga, Alberto Termine, Martin Gjoreski, Giuseppe Marra, and Marc Langheinrich. Causal concept graph models: Beyond causal opacity in deep learning. *arXiv preprint arXiv:2405.16507*, 2024.

Jessica Echterhoff, An Yan, Kyungtae Han, Amr Abdelraouf, Rohit Gupta, and Julian McAuley. Driving through the concept gridlock: Unraveling explainability bottlenecks in automated driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 7346–7355, 2024.

Asma Ghandeharioun, Been Kim, Chun-Liang Li, Brendan Jou, Brian Eoff, and Rosalind Picard. DISSECT: Disentangled simultaneous explanations via concept traversals. In *Proceedings of the 10th International Conference on Learning Representations (ICLR)*, 2022.

Amirata Ghorbani, James Wexler, James Zou, and Been Kim. Towards automatic concept-based explanations. In *Proceedings of the 33rd International Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Mara Graziani, Vincent Andrearczyk, and Henning Müller. Regression concept vectors for bidirectional explanations in histopathology. In *International Workshop on Interpretability of Machine Intelligence in Medical Image Computing (IMIMIC)*, pp. 124–132, 2018.

Balint Gyevnar, Cheng Wang, Christopher G. Lucas, Shay B. Cohen, and Stefano V. Albrecht. Causal explanations for sequential decision-making in multi-agent systems. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2024.

Peter Heidlauf, Alexander Collins, Michael Bolender, and Stanley Bak. Verification challenges in F-16 ground collision avoidance and other automated maneuvers. In *Proceedings of the 5th International Workshop on Applied Verification of Continuous and Hybrid Systems (ARCH)*, 2018.

Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. Transforming auto-encoders. In *Proceedings of the 21st International Conference on Artificial Neural Networks (ICANN)*, 2011.

Kosuke Imai, Luke Keele, and Dustin Tingley. A general approach to causal mediation analysis. *Psychological Methods*, 15(4):309, 2010.

Dmitry Kazhdan, Botty Dimanov, Helena Andres Terre, Mateja Jamnik, Pietro Liò, and Adrian Weller. Is disentanglement all you need? Comparing concept-based and disentanglement approaches. In *ICLR 2021 Workshop on Weakly Supervised Learning*, 2021.

Eoin M Kenny, Mycal Tucker, and Julie Shah. Towards interpretable deep reinforcement learning with human-friendly prototypes. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*, 2023.

Eoin M Kenny, Akshay Dharmavaram, Sang Uk Lee, Tung Phan-Minh, Shreyas Rajesh, Yunqing Hu, Laura Major, Momchil S Tomov, and Julie A Shah. Explainable deep learning improves human mental models of self-driving cars. *arXiv preprint arXiv:2411.18714*, 2024.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

Michael Lanier, Ying Xu, Nathan Jacobs, Chongjie Zhang, and Yevgeniy Vorobeychik. Learning interpretable policies in hindsight-observable pomdps through partially supervised reinforcement learning. In *Proceedings of the International Conference on Machine Learning and Applications (ICMLA)*, 2024.

Hopin Lee, Aidan G Cashin, Sarah E Lamb, Sally Hopewell, Stijn Vansteelandt, Tyler J VanderWeele, David P MacKinnon, Gemma Mansell, Gary S Collins, Robert M Golub, et al. A guideline for reporting mediation analyses of randomized trials and observational studies: the agrema statement. *Journal of the American Medical Association*, 326(11):1045–1056, 2021.

Mingxuan Li, Junzhe Zhang, and Elias Bareinboim. Confounding robust deep reinforcement learning: A causal approach. In *Proceedings of the 40th International Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2025a. (to appear).

Peilang Li, Umer Siddique, and Yongcan Cao. From explainability to interpretability: Interpretable policies in reinforcement learning via model explanation. In *AAAI 2025 Workshop on Deployable AI (DAI)*, 2025b.

Wen Wei Loh, Beatrijs Moerkerke, Tom Loeys, and Stijn Vansteelandt. Disentangling indirect effects through multiple mediators without assuming any causal structure among the mediators. *Psychological Methods*, 27(6):982, 2022.

Charles Lovering, Jessica Zosa Forde, George Konidaris, Ellie Pavlick, and Michael Littman. Evaluation beyond task performance: Analyzing concepts in AlphaZero in Hex. In Alice H Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Proceedings of the 36th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Proceedings of the 31th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

David P MacKinnon. *Introduction to Statistical Mediation Analysis*. Lawrence Erlbaum Associates, 2008. ISBN 9780805839746.

Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. Explainable reinforcement learning through a causal lens. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

Saaduddin Mahmud, Sandhya Saisubramanian, and Shlomo Zilberstein. Verification and validation of AI systems using explanations. In *Proceedings of the AAAI Symposium on AI Trustworthiness and Risk Assessment for Challenging Contexts (ATRACC)*, 2024a.

Saaduddin Mahmud, Marcell Vazquez-Chanlatte, Stefan Witwicki, and Shlomo Zilberstein. Explaining the behavior of POMDP-based agents through the impact of counterfactual information. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2024b.

Samer B Nashed, Saaduddin Mahmud, Claudia V Goldman, and Shlomo Zilberstein. Causal explanations for sequential decision making. *Journal of Artificial Intelligence Research*, 83(17), 2025.

Matthew L Olson, Roli Khanna, Lawrence Neal, Fuxin Li, and Weng-Keen Wong. Counterfactual state explanations for reinforcement learning agents via generative deep learning. *Artificial Intelligence*, 295:103455, 2021.

Daniel Omeiza, Helena Webb, Marina Jirotka, and Lars Kunze. Explanations in autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):10142–10162, 2021.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition, 2009. ISBN 052189560X.

Judea Pearl. The causal mediation formula – A guide to the assessment of pathways and mechanisms. *Prevention Science*, 13(4):426–436, 2012.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press, 2017. ISBN 0262037319.

Tung Phan-Minh, Forbes Howington, Ting-Sheng Chu, Momchil S Tomov, Robert E Beaudoin, Sang Uk Lee, Nanxiang Li, Caglayan Dicle, Samuel Findler, Francisco Suarez-Ruiz, Bo Yang, Sammy Omari, and Eric M Wolff. DriveIRL: Drive in real life with inverse reinforcement learning. In *Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

Adrian P Pope, Jaime S Ide, Daria Mićović, Henry Diaz, Jason C Twedt, Kevin Alcedo, Thayne T Walker, David Rosenbluth, Lee Ritholtz, and Daniel Javorsek. Hierarchical reinforcement learning for air combat at DARPA's AlphaDogfight trials. *IEEE Transactions on Artificial Intelligence*, 4(6):1371–1385, 2022.

Yilue Qian, Peiyu Yu, Ying Nian Wu, Yao Su, Wei Wang, and Lifeng Fan. Learning concept-based causal transition and symbolic reasoning for visual planning. In *Proceedings of the 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2016.

Joshua Rountree, Patrick Hipelius, Brian Dienst, Jonathan Aronoff, Ryan Neely, Robert Steigerwald, Skylar Griffis, David de Schweinitz, Chiawei Lee, and Ryan Hefron. Testing artificial intelligence in high-performance, tactical aircraft. In *Proceedings of the 2021 IEEE Aerospace Conference (50100)*, 2021.

Kangrui Ruan, Junzhe Zhang, Xuan Di, and Elias Bareinboim. Causal imitation learning via inverse reinforcement learning. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*, 2023.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, May 2019.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. *Proceedings of the 31th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Lindsay Sanneman and Julie A Shah. The situation awareness framework for explainable AI (SAFE-AI) and human factors considerations for XAI systems. *International Journal of Human–Computer Interaction*, 38(18-20):1772–1788, 2022.

Lindsay Sanneman, Mycal Tucker, and Julie A Shah. An information bottleneck characterization of the understanding-workload tradeoff in human-centered explainable AI. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Oliver Schulte and Pascal Poupart. When should reinforcement learning use causal reasoning? *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856.

Lisa Schut, Nenad Tomašev, Thomas McGrath, Demis Hassabis, Ulrich Paquet, and Been Kim. Bridging the human–AI knowledge gap through concept discovery and transfer in AlphaZero. *Proceedings of the National Academy of Sciences*, 122(13):e2406675122, 2025.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. SmoothGrad: Removing noise by adding noise. In *ICML 2017 Workshop on Visualization for Deep Learning*, 2017.

Oswin So and Chuchu Fan. Solving stabilize-avoid optimal control via epigraph form and deep reinforcement learning. In *Proceedings of Robotics: Science and Systems (R:SS)*, 2023.

Conner Sprague, Eric B Wendoloski, and Ingrid Guch. Interpretable AI for deep learning-based meteorological applications. In *American Meteorological Society Meeting Abstracts*, volume 99, pp. TJ17–5, 2019.

Sarath Sreedharan, Anagha Kulkarni, and Subbarao Kambhampati. *Explainable Human-AI Interaction: A Planning Perspective*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2022. ISBN 9781636392905.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

Andong Tan, Fengtao Zhou, and Hao Chen. Explain via any concept: Concept bottleneck model with open vocabulary concepts. In *Proceedings of the 18th European Conference on Computer Vision (ECCV)*, 2024.

Chen Tang, Ben Abbatematteo, Jiaheng Hu, Rohan Chandra, Roberto Martín-Martín, and Peter Stone. Deep reinforcement learning for robotics: A survey of real-world successes. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI)*, 2025.

Nicholas Tomlin, Andre He, and Dan Klein. Understanding game-playing agents with natural language annotations. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022.

Stratis Tsirtsis, Abir De, and Manuel Rodriguez. Counterfactual explanations in sequential decision making under uncertainty. *Proceedings of the 35th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Mycal Tucker, Roger Levy, Julie A Shah, and Noga Zaslavsky. Trading off utility, informativeness, and complexity in emergent communication. In *Proceedings of the 36th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

United States Air Force. Flying operations: T-38C flying fundamentals. Manual AETCMAN 11-251, Department of Defense, San Antonio, TX, September 2024. URL https://static.e-publishing.af.mil/production/1/aetc/publication/aetcman11-251/aetcman11-251.pdf. 10 September 2024.

United States Air Force. Flying operations: T-6 primary flying. Manual AETCMAN 11-248, Department of Defense, San Antonio, TX, August 2025. URL https://static.e-publishing.af.mil/production/1/aetc/publication/aetcman11-248/aetcman11-248.pdf. 13 August 2025.

Tyler J VanderWeele. *Explanation in Causal Inference: Methods for Mediation and Interaction*. Oxford University Press, 2015. ISBN 9780199325870.

Stijn Vansteelandt and Rhian M Daniel. Interventional effects for mediation analysis with multiple mediators. *Epidemiology*, 28(2):258–265, 2017.

Stijn Vansteelandt and Tyler J VanderWeele. Natural direct and indirect effects on the exposed: Effect decomposition under weaker assumptions. *Biometrics*, 68(4):1019–1027, 2012.

Pulkit Verma and Julie A Shah. Interpretability analysis of symbolic representations for sequential decision-making systems. In *HRI 2025 Workshop on Explainability for Human-Robot Collaboration: Real-World Concerns (X-HRI)*, 2025.

Pulkit Verma and Siddharth Srivastava. Learning causally accurate models for autonomous assessment of deterministic black-box agents. Technical Report TR-ASUSCAI-2024-001, 2024.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology*, 31 (2):841–888, 2017.

Zizhao Wang, Xuesu Xiao, Zifan Xu, Yuke Zhu, and Peter Stone. Causal dynamics learning for task-independent state abstraction. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 2022.

Tom Ward. The US Air Force is moving fast on AI-piloted fighter jets. *WIRED*, March 2023.

Ruyi Yang, Jingyu Hu, Zihao Li, Jianli Mu, Tingzhao Yu, Jiangjiang Xia, Xuhong Li, Aritra Dasgupta, and Haoyi Xiong. Interpretable machine learning for weather and climate prediction: A review. *Atmospheric Environment*, 338:120797, 2024.

Hugo Yeche, Justin Harrison, and Tess Berthier. UBS: A dimension-agnostic metric for concept vector interpretability applied to radiomics. In *Proceedings of the 22nd International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2019.

Chih-Kuan Yeh, Been Kim, Sercan Ö. Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. In *Proceedings of the 34th Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Maryam Zare, Parham M Kebria, Abbas Khosravi, and Saeid Nahavandi. A survey of imitation learning: Algorithms, recent developments, and challenges. *IEEE Transactions on Cybernetics*, 2024.

Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. Do feature attribution methods correctly attribute features? In *Proceedings of the 36th AAAI conference on Artificial Intelligence (AAAI)*, 2022.

A Extended Related Work

In this section, we discuss a few orthogonal research directions and explain how our approach differs from them.

Causal Attribution Methods in Deep Learning Concept-based explanations have been adapted to control and planning including learning joint embeddings between state-action pairs and explanations (Das et al., 2023), gradient and path-based scored (Sundararajan et al., 2017), concept level testing (Kim et al., 2018), and self-explaining networks (Achtibat et al., 2023). Further work refines concept attribution and completeness guarantees (Yeh et al., 2020; Bai et al., 2023) and attributes relevance to produce human-understandable rationales (Achtibat et al., 2023). However, these approaches are correlational. In safety and task-critical autonomy, such correlational explanations can be misleading. Saliency or concept relevance can be high even when a factor is not on a causal path to the action or decision (Zhou et al., 2022). CCW-Net instead targets causal quantities. Our approach differs by explicitly grounding attribution in estimated causal effects. We not only compute how actions respond to concept changes, but also check these sensitivities against interventional baselines drawn from data.

Interpretability of Sequential Decision Making Systems A large body of work endeavors to make sequential decision-making systems more interpretable (Sreedharan et al., 2022; Lanier et al., 2024; Li et al., 2025b; Verma & Shah, 2025). Post-hoc approaches explain behaviors via counterfactuals (Tsirtsis et al., 2021; Olson et al., 2021) and concept bottleneck models (Koh et al., 2020; Delfosse et al., 2024) to improve user understanding. In safety critical domains such as autonomous driving, surveys highlight gaps between correlational attributions and user needs for actionable, task-grounded explanations (Omeiza et al., 2021; Atakishiyev et al., 2025). The field further emphasizes balancing informativeness with cognitive load (Sanneman et al., 2024). CCW-Net advances this field by grounding concept-level explanations in causal effects on actions, rather than correlations, while providing information rich, real-world domain explanations.

Causal Explanations in Sequential Decision making Counterfactual reasoning is an accepted approach for explanations (Wachter et al., 2017) and has been explored for sequential decision making policies in multi-agent settings (Gyevnar et al., 2024), simple deterministic settings (Verma & Srivastava, 2024), partially-observable Markov decision processes (POMDPs) through counterfactional information impact (Mahmud et al., 2024b), and recent surveys outlining causal explanation desiderata for sequential tasks (Nashed et al., 2025). Recent work also studies causally reliable concept bottlenecks (De Felice et al., 2025); however, our work goes beyond this by imposing causal relationships in within the policy.

Causal reinforcement learning: Causal approaches have been used to mitigate confounding in reinforcement learning (RL) (Li et al., 2025a), formulate causal imitation learning via inverse reinforcement learning (IRL) (Ruan et al., 2023), and to identify when causal reasoning benefits RL (Schulte & Poupart, 2025). CCW-Net contributes to this field by performing mediation analysis on demonstrations to infer concept-level causal targets, then use them to shape the policy's internal concept-to-action attribution during imitation learning. CCW-Net requires no explicit environment structural causal model and applies across any differentiable policy head.

Discovering causal relationships in CBMs Recent work aims to reveal existing causal structure among concepts or endow concept models with causal meaning (Dominici et al., 2024; De Felice et al., 2025). Our work compliments this such that instead of discovering a causal concept graph, we enhance the representation capacity of concept to carry both presence and context-dependent expression, and causally align concepts to actions by matching interventional causal targets.

B VARIABLE DICTIONARY

Symbol	Definition
X	Raw observed features
Z	Frozen backbone latent
C	Covariates used for matching (use X if available, else Z)
A	Treatment (binary concept activations per concept pair)
$M = (c_1, \ldots, c_J)$	Mediators: vector concept representations after normalization
Y	Outcome (expert action)
s_j	Pre-activation concept block for concept j
ψ_j	Per-concept normalization (softmax, squash, $\tanh$/identity)
$c_j = \psi_j(s_j)$	Concept representation consumed by g_θ
g_θ	Policy head mapping concepts to actions
$\hat{Y} = g_\theta(c)$	Predicted action (includes output nonlinearity, e.g. $\tanh$)
$\Delta\hat{a}_j$	Local concept→action effect (Jacobian of $g_\theta \circ \psi$ in block j)
u_j^s, u_j^c	Radial unit directions in s-space and induced direction in c-space
$m_j(x)$	Activity mask for concept j (1 if active for x, else 0)
IE_j	Interventional indirect effect of concept j in action space
TE, IE_μ	Total effect; residual interaction ($IE_\mu = TE - \sum_j IE_j$)
ϕ_μ	Signed interaction share $\langle IE_\mu, TE\rangle / \|TE\|^2$
$\mathcal{R}_{(L,C)}$	Treatment-group reference set for bundle (L, C)
$Y_\theta^{\text{frozen}}(M)$	Action from forwarding mediators M through the frozen decoder
D_i	Standardized squared distance in C for matching
w_i	Kernel weight for candidate i (Gaussian with bandwidth τ)
δ, τ, S	Caliper, kernel temperature, Monte Carlo draws
ESS	Effective sample size $= 1/\sum_i w_i^2$ (per query; averaged in reports)
coverage	Fraction of queries with at least one eligible neighbor (under caliper)
accepts/sample	Mean number of eligible neighbors per query (under caliper)
fallback rate	Fraction of queries with no eligible neighbors (nearest-neighbor fallback)
$\lambda_{\text{cls}}, \lambda_{\text{align}}$	Loss weights for concept supervision and alignment

Table 1: Variable dictionary.

C ADDITIONAL DOMAIN INFORMATION

C.1 AIRCRAFT FORMATION TASK

Extended trail formation flight is a real-world, complex task used in formal advanced pilot training to teach pilots how to manage aircraft position and energy with respect to another aircraft. Lead pursuit, lag pursuit, climb, and dive (represented as $\{Lead, Lag, Climb, Dive\}$ concepts in CCW-Net) are are well grounded in human-interpretability as they are formally used to teach, communicate, fly, and debrief the extended trail formation task.

C.1.1 EXTENDED TRAIL TASK

Extended trail consists of a formation of two aircraft: a lead and a chase aircraft. The chase aircraft's task is to remain within a defined cone behind the lead aircraft. The extended trail cone is defined as 30° to 45° off of the lead aircraft's tail and between 500 feet and 1500 feet behind the lead aircraft (Figure 2).

C.2 POLICY TRAINING

The expert policy is trained using the method from So & Fan (2023), a variant of PPO (Schulman et al., 2017) that additionally considers per-timestep safety constraints. The original aircraft environment (Heidlauf et al., 2018; So & Fan, 2023) with a 4-dimensional control space consisting of the desired load factor, desired roll rate, desired yaw rate, and throttle, and a 20-dimensional observation space consisting of the states of the two aircraft and the current control. During training, we fix the throttle and set the desired yaw rate to 0. Moreover, to improve the temporal smoothness of

the controls, we control the change in the control outputs and store the current controls in the state. The training framework is implemented in JAX (Bradbury et al., 2018) in the JAX version of the aircraft environment (So & Fan, 2023).

C.3 Extended Trail Concepts

Four concepts are used to teach, communicate, fly, and debrief the extended trail task: Lead pursuit, lag pursuit, climb, and dive (United States Air Force, 2025) (Figure 2). Lead and lag pursuit are fundamental maneuvers and application to flight throughout aviation. Lead pursuit is used to catch up with the lead aircraft while lag pursuit is used to increase distance from the lead aircraft. Lead pursuit is performed by pointing the nose of the chase aircraft ahead of the lead aircraft. Conversely, lag pursuit is performed by pointing the nose of the chase aircraft behind the lead aircraft. Climb and dive are conceptually simpler in that they are performed by ascending or descending in altitude.

There are an infinite amount of ways to perform lead pursuit, lag pursuit, climb and dive in two-dimensional action space (pitch and roll). For example, a dive may be performed while aircraft is at any roll angle (e.g., upright, inverted, or any angle in between) and at varying intensity (e.g., shallow versus steep dives). Similarly, lead and lag pursuit can be performed in a number of ways. It is possible for a given action input to accommodate any of the four concepts. As such, concept explanations based on action space are insufficient to describe these concepts which human pilots readily understand stressing the need for higher-level, abstract concept representation.

Notably, $\{Lead, Lag\}$ and $\{Climb, Dive\}$ are physically mutually exclusive within each subset (e.g., an aircraft cannot simultaneously climb and dive at the same time).

D CCW-Net Training

D.1 Training Summary

We train CCW-Net on 500,000 trajectory state-action (s, a) samples with concept labels. Labeled samples (with concept pairs) are split into train/val/test (train 60%, test 15%, validation 15%) with stratification over the joint concept label. The pretrained backbone encoder is frozen. We train only the concept module h_θ and the policy head g_θ. Training runs for 100 epochs. A warm-up of 50 epochs uses imitation and concept supervision; the causal alignment is activated after warm-up. We use Adam optimizer, learning rate 1×10^{-4}, batch size 256, five random seeds per setting. We minimize $\mathcal{L} = \mathcal{L}_{\text{imit}} + \lambda_{\text{cls}} \mathcal{L}_{\text{cls}} + \lambda_{\text{align}} \mathcal{L}_{\text{align}}$, where $\lambda_{\text{align}} = 0$ during warm-up. Concept supervision: margin loss on capsule lengths for the capsule network (Appendix F.1; two-way cross-entropy per concept pair for CBM and attention implementations (Appendix F.2 and F.3.

Interventional estimation (Monte Carlo). Treatment groups are the concept-pair bundles $(L, C) \in \{\text{Lead,Lag}\} \times \{\text{Climb,Dive}\}$. Per group we store (C_i, M_i) and standardization (μ, σ). Matching uses standardized squared distance with caliper $\delta = 1.5$ and Gaussian-kernel weights with temperature $\tau = 0.25$; donors are sampled via the Gumbel–Max trick with nearest-neighbor fallback. We use $S = 8$ draws per swap and compute IE_j, TE (and IE_μ) with the frozen decoder saved in the reference sets. Reference sets are cross-fitted (two folds), first built after epoch 2 (given warm-up ≥ 10 epochs), and then refreshed every epoch via partial replacement fraction $p = 0.1$ to smooth distributional drift.

Diagnostics and reporting. We report test MSE, concept accuracies, mean active alignment $\cos(\Delta\hat{a}_j, \text{IE}_j)$, per-concept cosine, and the signed interaction share $\phi_\mu = \langle \text{IE}_\mu, \text{TE} \rangle / \|\text{TE}\|^2$. Overlap diagnostics for matching include accepts/sample, coverage, effective sample size (ESS), standardized distance quantiles, and fallback rate.

Table 2: CCW-Net training settings (shared across architectures unless noted).

Item	Setting
Training set size	500,000 samples
Epochs / Warm-up	100 / 50 (alignment off during warm-up)
Batch size / Optimizer / LR	256 / Adam / 1×10^{-4}
Seeds per setting	5
Frozen vs. trained	Backbone frozen; train h_θ and g_θ
Concept supervision	Capsules: margin ($m_+ = 0.9$, $m_- = 0.1$); CBM/Attn: cross-entropy per pair
Alignment loss	Masked cosine on active concepts only
Jacobian	$u_j^s = s_j/(\|s_j\| + \varepsilon)$ (one Jacobian per concept)
MC draws / Matching	$S = 8$; caliper $\delta = 1.5$, kernel $\tau = 0.25$, Gumbel–Max sampling
Reference refresh	Cross-fitted; partial replacement $p = 0.1$ each refresh (every epoch)
Checkpoints	Every 5 epochs; final at epoch 100
Reported metrics	Test MSE, Acc-LL, Acc-CD, mean active cosine, per-concept cosine, ϕ_μ

D.2 Training Curves

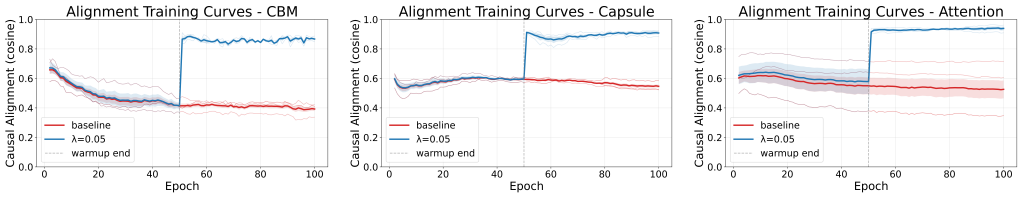


Figure 5: Causal training curves for the concept bottleneck model (CBM), capsule network, and attention architectures. Causal losses applied at epoch 50.

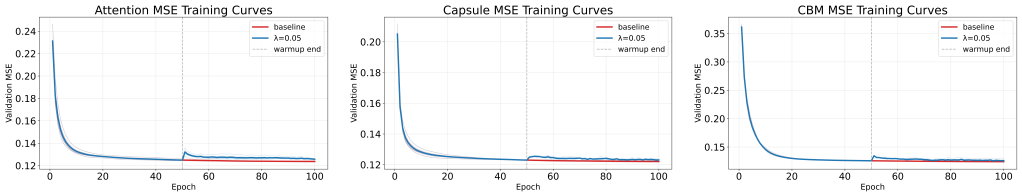


Figure 6: Imitation training curves (mean squared error - MSE) for the concept bottleneck model (CBM), capsule network, and attention architectures. Causal losses applied at epoch 50.

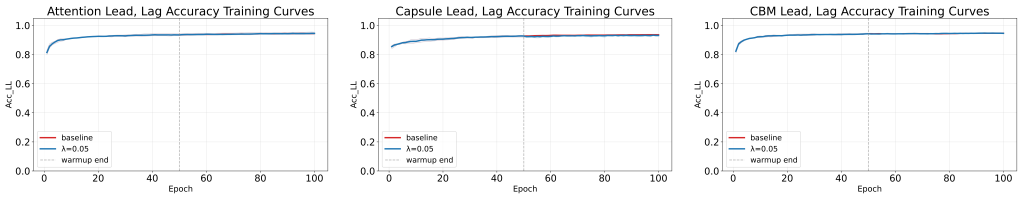


Figure 7: Lead/Lag pursuit concept classification training curves for the concept bottleneck model (CBM), capsule network, and attention architectures. Causal losses applied at epoch 50.

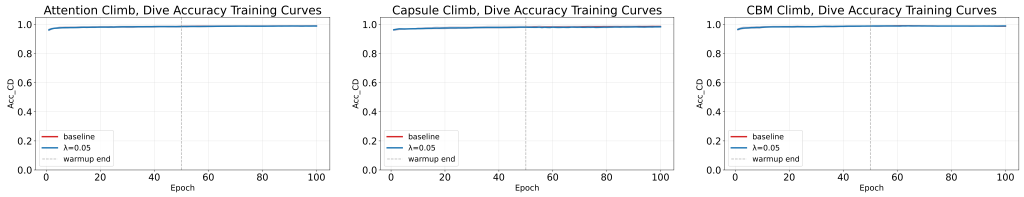


Figure 8: Climb/Dive concept classification training curves for the concept bottleneck model (CBM), capsule network, and attention architectures. Causal losses applied at epoch 50.

D.3 Data Summary

D.3.1 Statistical Summary

Table 3: Performance and causal alignment by architecture at $\lambda_{\text{align}} = 0.05$ (mean±95% CI). Δ rows report Causal−Baseline with significance from paired tests.

Architecture	Regime	Test MSE ↓	Acc_LL ↑	Acc_CD ↑	Mean Causal Alignment ↑
CBM	Baseline	0.122±0.002	0.950±0.005	0.991±0.003	0.363±0.057
	Causal (λ=0.05)	0.125±0.003	0.950±0.011	0.990±0.006	0.857±0.027
	Δ (C−B)	+0.002 ns	+0.000 ns	−0.001 ns	+0.449 **
Capsule	Baseline	0.121±0.002	0.944±0.006	0.984±0.002	0.549±0.005
	Causal (λ=0.05)	0.124±0.003	0.941±0.010	0.983±0.002	0.918±0.015
	Δ (C−B)	+0.003 *	−0.003 ns	−0.001 ns	+0.368 ***
Attention	Baseline	0.123±0.002	0.944±0.005	0.990±0.003	0.449±0.186
	Causal (λ=0.05)	0.124±0.005	0.946±0.006	0.989±0.006	0.939±0.026
	Δ (C−B)	+0.001 ns	+0.001 ns	+0.001 ns	+0.234 ns

Note: Significance levels: *** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$, ns = not significant. Acc_LL = Lead/Lag accuracy, Acc_CD = Climb/Dive accuracy, Mean Causal Alignment = mean active cosine. Causal alignment is cosine similarity in $[-1, 1]$. Accuracies are proportions in $[0, 1]$. Values are mean ± 95% CI across seeds ($n = 3$). All tests use paired samples across matched seeds.

D.3.2 Residual interaction share.

We report residual interaction share ϕ_μ per architecture with and without causal losses applied. This represents the fraction of a concept's total causal effect on the action that cannot be attributed to any single concept alone, but instead due to interactions between multiple concepts.

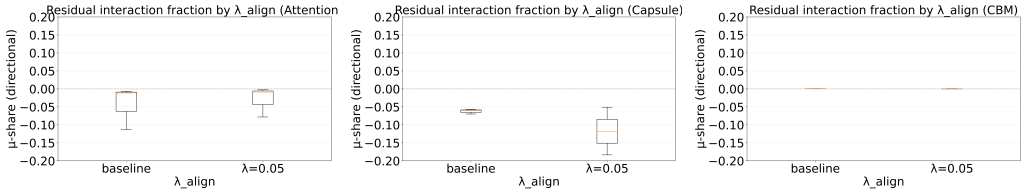


Figure 9: Residual interaction share across architectures with and without causal loss applied. Negative values imply effects against the total effect.

D.3.3 Overlap Diagnostics

We quantify the quality of kernel matching on covariates C used to form interventional draws with the following diagnostics, computed within each treatment group and then aggregated over samples and runs.

Coverage. Fraction of queries with at least one eligible neighbor under the caliper,

$$\text{coverage} \;=\; \frac{1}{B}\sum_{b=1}^{B}\mathbb{1}\Big[\sum_{i}\mathbb{1}[D_{bi}\le\delta^2]>0\Big],$$

where $\mathbb{1}$ is the indicator function (i.e., holds value 1 when the condition inside the brackets is true and 0 otherwise), D_{bi} is the standardized squared distance from query b to candidate i, δ is the caliper, and B is the number of queries.

Accepts per sample. Mean number of eligible neighbors per query under the caliper,

$$\text{accepts/sample} \;=\; \frac{1}{B}\sum_{b=1}^{B}\sum_{i}\mathbf{1}[D_{bi}\le\delta^2].$$

Effective sample size (ESS). For kernel weights w_{bi} normalized over eligible neighbors of query b,

$$\text{ESS} \;=\; \frac{1}{B}\sum_{b=1}^{B}\frac{1}{\sum_i w_{bi}^2}.$$

Fallback rate. Fraction of queries that required a nearest-neighbor fallback because no neighbor met the caliper,

$$\text{fallback rate} \;=\; \frac{\#\{\text{queries with zero eligible neighbors}\}}{B}.$$

Distance quantiles. Summary of standardized squared distances among eligible neighbors: minimum, median, and 90th percentile. These describe the tightness of the matching neighborhood in C.

All metrics are computed per run, then summarized as mean $\pm$ 95% CI across runs.

Why present these diagnostics. These quantities directly assess the overlap (positivity) needed for identifying interventional effects from observational data and indicate the stability of the Monte-Carlo estimates used for IE_j and TE. Because matching operates only on C, we expect similar overlap statistics across alignment weights; we therefore report architecture-level summaries and include baseline and weight breakdowns for completeness.

Table 4: Overlap diagnostics by architecture and training regime (baseline vs. causal).

Architecture	Regime	Accepts/sample	Coverage	ESS	Fallback rate (%)
CBM	Baseline	0.426±0.004	0.984±0.002	230.270±3.980	0.031±0.004
CBM	Causal	0.428±0.006	0.986±0.002	231.356±8.288	0.028±0.004
Capsule	Baseline	0.427±0.003	0.985±0.002	229.222±3.507	0.030±0.003
Capsule	Causal	0.428±0.006	0.986±0.002	231.356±8.288	0.028±0.004
Attention	Baseline	0.427±0.003	0.985±0.002	229.744±4.359	0.031±0.004
Attention	Causal	0.428±0.006	0.986±0.002	231.356±8.288	0.028±0.004

Note: Accepts/sample = mean eligible neighbors under the caliper; Coverage = fraction with at least one eligible neighbor; ESS = effective sample size $1/\sum_i w_i^2$ averaged over queries; Fallback rate = fraction requiring nearest-neighbor fallback because no eligible neighbors were found under the caliper. Values are mean $\pm$ 95% CI across runs.

Interpretation. High coverage and low fallback indicate adequate support in C for the counterfactual draws used in interventional estimation. Larger accepts/sample and ESS reflect richer local neighborhoods and lower variance in the Monte-Carlo estimates. Distance quantiles closer to zero indicate tighter matches under the caliper. Given that matching is performed soley on C, consistency of these metrics across architectures and alignment weights is expected.

E Interventional Estimation and Alignment Details

E.1 Notation and Setup

We represent each logged expert trajectory sample with concept labels as (C, A, M, Y) with covariates C, treatment A (binary concept activations per axis), mediators $M = (c_1, \ldots, c_J)$ (with c_j each as vector concept representations), and outcome Y (expert action). In our experiments $A = (L, C)$ where $L \in \{\text{Lead}, \text{Lag}\}$ and $C \in \{\text{Climb}, \text{Dive}\}$; let $\bar{L}$ and $\bar{C}$ denote their complements. During training we maintain treatment-group reference sets $\{\mathcal{R}_{(L,C)}\}$ that store tuples (C_i, M_i) and per-group standardization statistics $(\mu_{(L,C)}, \sigma_{(L,C)})$. Let $Y_\theta^{\text{frozen}}(M)$ denote the action from forwarding mediators M through the frozen decoder saved with the reference sets.

E.2 Reference Sets and Matching

Given a query covariate vector C^* and observed treatment $a_{\text{obs}} = (L_{\text{obs}}, C_{\text{obs}})$, we approximate draws from the conditional mediator distribution within the relevant group(s) via kernel matching with a caliper and Gumbel–Max sampling:

1. Standardize within group: $\tilde{C} = (C - \mu_{(L,C)})/\sigma_{(L,C)}$.
2. Distance and caliper: $D_i = \|\tilde{C}^* - \tilde{C}_i\|_2^2/p$, keep candidates with $D_i \le \delta^2$.
3. Kernel weights: $w_i \propto \exp\big(-D_i/(2\tau^2)\big)$, normalized over eligible neighbors.
4. Sampling: draw indices with the Gumbel–Max trick $\arg\max_i\{\log w_i + G_i\}$ (nearest-neighbor fallback if no eligible neighbors).

We monitor coverage (fraction with at least one neighbor), accepts per sample (mean eligible neighbors), effective sample size, and fallback rate during training.

E.2.1 Monte Carlo Sampling for Interventional Draws

Given a query covariate vector C^* and an interventional target treatment a', we approximate draws from the conditional mediator distribution $P(M \mid A = a', C = C^*)$ using calipered kernel matching and the Gumbel–Max trick. This produces donor mediators that are close in covariate space while allowing a smooth, vectorized sampler.

Eligible set under a caliper. Within treatment group a', we standardize covariates using $(\mu_{a'}, \sigma_{a'})$ and compute

$$\tilde{C}_i \;=\; \frac{C_i - \mu_{a'}}{\sigma_{a'}}, \qquad \tilde{C}^* \;=\; \frac{C^* - \mu_{a'}}{\sigma_{a'}},$$

then define per-candidate standardized squared distances

$$D_i \;=\; \frac{\|\tilde{C}^* - \tilde{C}_i\|_2^2}{p}.$$

We keep candidates within a caliper δ as the eligible set

$$E \;=\; \big\{i \,:\, D_i \le \delta^2\big\}.$$

Kernel weights. On the eligible set E, define Gaussian-kernel weights

$$w_i \;\propto\; \exp\Big(-\frac{D_i}{2\tau^2}\Big), \qquad \sum_{i\in E} w_i \;=\; 1,$$

where τ is the kernel bandwidth. If $E = \varnothing$, we fall back to the nearest neighbor $i^\star = \arg\min_i D_i$ and set $E = \{i^\star\}$, $w_{i^\star} = 1$.

Gumbel–Max sampling (vectorized). To draw S i.i.d. indices from the categorical distribution over E, we use the Gumbel–Max trick:

$$i^{(s)} \;=\; \arg\max_{i\in E}\Big\{\log w_i \;+\; G_i^{(s)}\Big\}, \qquad G_i^{(s)} \sim \text{Gumbel}(0,1), \quad s = 1, \ldots, S.$$

This is equivalent to multinomial sampling but is easily batched and JAX-friendly. Each sampled index $i^{(s)}$ yields a donor mediator set $M^{(s)}$.

Independent draws for two concept pairs. When we intervene on both concept pairs to form Y_{both}, we draw donors *independently* from the two complementary treatment groups determined by the observed pair: one from $(\bar{L},\, C_{\text{obs}})$ and one from $(L_{\text{obs}},\, \bar{C})$ (each with its own calipered kernel and Gumbel–Max). This preserves the intended factorization in our hybrid construction.

Monte Carlo estimator and variance. For each target treatment a' we average the frozen-decoder outputs over S sampled donors,

$$\hat{Y}(a') \;=\; \frac{1}{S}\sum_{s=1}^{S} Y_\theta^{\text{frozen}}\big(M^{(s)} \mid A = a',\, C = C^*\big),$$

and use these $\hat{Y}(\cdot)$ to compute IE_j, TE, and IE_μ as in the main text. We report effective sample size (ESS) and coverage diagnostics to characterize variance and support; larger accepts/sample and ESS generally imply lower Monte Carlo variance.

Reproducibility and efficiency. Sampling is seeded per epoch and minibatch (key splitting and folding), yielding deterministic reruns per configuration. Distances and Gumbel–Max are fully vectorized across queries and candidate sets. When reference sets are refreshed, we use partial replacement to avoid large distributional jumps while keeping overlap statistics healthy.

E.3 Counterfactual Forward Passes ("What If" Analysis)

For a sample with observed treatment $a_{\text{obs}} = (L_{\text{obs}}, C_{\text{obs}})$ and complements $\bar{L}, \bar{C}$, define the observed and interventional outcomes by forwarding mediator hybrids through the frozen decoder:

$$Y_{\text{obs}} := Y_\theta^{\text{frozen}}(M_{\text{obs}}), \tag{1}$$

$$Y_{L\leftarrow\bar{L}} := \frac{1}{S}\sum_{s=1}^{S} Y_\theta^{\text{frozen}}\big(M^{(s)}_{L\leftarrow\bar{L},\, C=C_{\text{obs}}}\big), \quad M^{(s)}_{L\leftarrow\bar{L},\, C=C_{\text{obs}}} \sim \mathcal{R}_{(\bar{L},\, C_{\text{obs}})}, \tag{2}$$

$$Y_{C\leftarrow\bar{C}} := \frac{1}{S}\sum_{s=1}^{S} Y_\theta^{\text{frozen}}\big(M^{(s)}_{C\leftarrow\bar{C},\, L=L_{\text{obs}}}\big), \quad M^{(s)}_{C\leftarrow\bar{C},\, L=L_{\text{obs}}} \sim \mathcal{R}_{(L_{\text{obs}},\, \bar{C})}, \tag{3}$$

$$Y_{\text{both}} := \frac{1}{S}\sum_{s=1}^{S} Y_\theta^{\text{frozen}}\big(M^{(s)}_{L\leftarrow\bar{L},\, C\leftarrow\bar{C}}\big), \quad \text{with independent draws from } \mathcal{R}_{(\bar{L},\, C_{\text{obs}})} \text{ and } \mathcal{R}_{(L_{\text{obs}},\, \bar{C})}. \tag{4}$$

E.4 Effect Decomposition

Define the total and axis-wise effects:

$$\text{TE} \;=\; Y_{\text{obs}} - Y_{\text{both}}, \qquad \Delta_L \;=\; Y_{\text{obs}} - Y_{L\leftarrow\bar{L}}, \qquad \Delta_C \;=\; Y_{\text{obs}} - Y_{C\leftarrow\bar{C}}. \tag{5}$$

Allocate per-concept interventional indirect effects from the axis-wise effects using the observed active concepts:

$$\text{IE}_{\text{Lead}} = \mathbb{1}[L_{\text{obs}} = \text{Lead}]\,\Delta_L, \qquad \text{IE}_{\text{Lag}} = \mathbb{1}[L_{\text{obs}} = \text{Lag}]\,\Delta_L, \tag{6}$$

$$\text{IE}_{\text{Climb}} = \mathbb{1}[C_{\text{obs}} = \text{Climb}]\,\Delta_C, \qquad \text{IE}_{\text{Dive}} = \mathbb{1}[C_{\text{obs}} = \text{Dive}]\,\Delta_C, \tag{7}$$

where $\mathbb{1}$ is the indicator function (i.e., holds value 1 when the condition inside the brackets is true and 0 otherwise). Define the residual interaction term:

$$\text{IE}_\mu \;=\; \text{TE} \;-\; \sum_j \text{IE}_j. \tag{8}$$

There is no direct effect by construction because the observation-to-action path is removed; all influence flows through mediators M.

Signed interaction share

$$\phi_\mu \;=\; \frac{\langle \mathrm{IE}_\mu,\, \mathrm{TE}\rangle}{\|\mathrm{TE}\|^2}, \tag{9}$$

which can be negative when interactions oppose the total effect.

E.5 Per-Concept Targets and Alignment Objective

For each active concept j, the interventional target is $\mathrm{IE}_j \in \mathbb{R}^A$. Let $\Delta\hat{a}_j$ denote the model's local per-concept effect (computed via a single Jacobian through the current head). We align $\Delta\hat{a}_j$ to IE_j with a masked cosine objective:

$$\mathcal{L}_{\text{align}}(x) \;=\; \sum_j m_j(x)\,\Big(1-\cos\big(\Delta\hat{a}_j(x),\, \mathrm{IE}_j(x)\big)\Big), \quad \cos(u,v)=\frac{\langle u,v\rangle}{\|u\|\,\|v\|+\varepsilon}, \tag{10}$$

where $m_j(x)\in\{0,1\}$ masks inactive concepts. The training loss is

$$\mathcal{L} \;=\; \mathcal{L}_{\text{imit}} \;+\; \lambda_{\text{cls}}\,\mathcal{L}_{\text{cls}} \;+\; \lambda_{\text{align}}\,\mathcal{L}_{\text{align}}, \tag{11}$$

with warmup epochs using $\lambda_{\text{align}}=0$.

E.6 Minimal Implementation Recipe

For each minibatch:

1. Compute $c=h_\theta(z)$ and $\hat{Y}=g_\theta(c)$.
2. For each concept j, compute $\Delta\hat{a}_j$ via a single Jacobian in a unit direction within block j.
3. Using $(L_{\text{obs}}, C_{\text{obs}})$ and C^*, draw S matched mediators from the relevant treatment-group reference set(s), form hybrids, and obtain IE_j, TE, and IE_μ via the frozen decoder.
4. Compute $\mathcal{L}_{\text{imit}}+\lambda_{\text{cls}}\,\mathcal{L}_{\text{cls}}+\lambda_{\text{align}}\,\mathcal{L}_{\text{align}}$.
5. Backpropagate into h_θ and g_θ; the decoder used for causal targets remains frozen.
6. Periodically refresh reference sets with cross-fitting; track coverage, ESS, and fallback rate.

F Policy Heads Evaluated for CCW-Net

High-level architecture. The concept module produces pre-activations $s=(s_1,\ldots,s_J)$ with $s_j\in\mathbb{R}^{d_j}$. Each concept block is normalized by a per-concept map $\psi_j:\mathbb{R}^{d_j}\to\mathbb{R}^{d_j}$ to yield the concept representation $c_j=\psi_j(s_j)$. The policy head $g_\theta:\mathbb{R}^{\sum_j d_j}\to\mathbb{R}^A$ maps $c=(c_1,\ldots,c_J)$ to the predicted action $\hat{Y}=g_\theta(c)$. CCW-Net's causal alignment uses a local per-concept effect $\Delta\hat{a}_j$ computed with a single Jacobian through the composed map $g_\theta\circ\psi$.

F.1 Capsule Network with Dynamic Routing

This is the first time that capsule networks have been implemented for sequential decision making in policy action decisions. Capsule networks naturally instantiate concepts as vectors where vector length represents presence or activation while vector orientation representations context-dependent expression of that concept.

Representation. Given $s\in\mathbb{R}^{J\times d}$, capsule squash normalizes each block

$$c_j=\psi_j(s_j)=\frac{\|s_j\|^2}{1+\|s_j\|^2}\,\frac{s_j}{\|s_j\|},$$

so $\|c_j\|\in(0,1)$ encodes presence and direction encodes expression.

Action mapping with routing. Votes $u_{jk} = W_{jk}c_j$ are routed to K action capsules using agreement-refined coefficients c_{jk}:

$$s_k = \sum_j c_{jk}\, u_{jk}, \qquad v_k = \text{squash}(s_k).$$

The action is

$$\hat{Y} = g_\theta(c) = \tanh\Big(\frac{1}{K}\sum_{k=1}^{K} P_k v_k\Big), \quad P_k \in \mathbb{R}^{d\times A}.$$

Concept supervision. Margin loss is applied to capsule lengths $\|c_j\|$ for labeled pairs.

Attribution for alignment. $\Delta\hat{a}_j$ is obtained by chaining the Jacobians of squash, the stored routing updates, the linear action maps, and the output $\tanh$, matching the implementation.

F.2 Vector Concept Bottleneck Model (CBM)

Here, we extend typical implementations of CBMs from concepts represented as scalars to concepts represented as vectors.

Representation. Given $s \in \mathbb{R}^{J\times d}$, each concept vector is block-normalized by a softmax

$$c_j = \psi_j(s_j) = \text{softmax}(s_j).$$

Action mapping. A per-concept linear action is summed and passed through $\tanh$:

$$\hat{Y} = g_\theta(c) = \tanh\Big(\sum_{j=1}^{J} P_j c_j + b\Big), \quad P_j \in \mathbb{R}^{d\times A},\ b \in \mathbb{R}^A.$$

Concept supervision. For each labeled concept pair, two logits are formed from block-means of s and trained with standard cross-entropy.

Attribution for alignment. With a radial direction in s-space $u_j^s = s_j/(\|s_j\|+\varepsilon)$, the local effect is

$$\Delta\hat{a}_j = J_{s_j}(g_\theta \circ \psi)(s)\, u_j^s = J_{c_j} g_\theta(c)\, u_j^c, \quad u_j^c = J_{s_j}\psi_j(s_j)\, u_j^s,$$

which corresponds to the softmax Jacobian composed with the linear output and the output $\tanh$.

F.3 Cross-Attention

Representation. Given $s \in \mathbb{R}^{J\times d}$, a non-capsule normalization yields

$$c_j = \psi_j(s_j) = \tanh(s_j).$$

Action mapping via attention. Each action dimension $a \in \{1,\dots,A\}$ attends to concepts using a learned query $q_a \in \mathbb{R}^d$ and key/value projections

$$k_j = K_{\text{proj}}\, c_j, \qquad v_j^{\text{val}} = V_{\text{proj}}\, c_j, \qquad \alpha_{aj} = \text{softmax}_j\Big(\frac{q_a^\top k_j}{\sqrt{d}}\Big).$$

With $\text{ctx}_a = \sum_j \alpha_{aj} v_j^{\text{val}}$, the action is

$$\hat{Y} = g_\theta(c) = \tanh\big(W_{\text{out}}\,\text{ctx} + b\big), \quad W_{\text{out}} \in \mathbb{R}^{A\times d},\ b \in \mathbb{R}^A.$$

where ctx is the context vector.

Concept supervision. As in CBM, two logits per labeled pair are derived from block-means of s and trained with cross-entropy.

Attribution for alignment. $\Delta\hat{a}_j$ is computed with a single Jacobian through the attention softmax and the output $\tanh$, using the induced radial direction u_j^c from u_j^s.