ABDUCTIVE LEARNING OF INTERMEDIATE VIDEO CONCEPTS IN A NEURO-SYMBOLIC AI ARCHITECTURE

Anonymous authors
Paper under double-blind review

ABSTRACT

A network trained to detect a compositional video event must discover the intermediate concepts that define it from examples alone; when the training data is narrow, it latches onto surface patterns of the training distribution rather than the event itself, and fails on unseen scenarios. Supervising those concepts directly fixes this but demands expensive per-frame, per-region annotation. We take the neuro-symbolic route instead: domain knowledge enters the model as differentiable logic rules and auxiliary symbolic induction losses. We couple a grid-adapted VideoMamba perception backbone with Scallop, a differentiable logic programming framework, and evaluate on a purpose-built synthetic testbed of role-based agents delivering instruments on a discrete grid, in which every intermediate concept has exact per-frame ground truth. With zero intermediate labels, the model reaches Pearson correlation above 0.95 on all three core channels on the training distribution; of these, only instrument possession is recovered without an appearance prior. On unseen scenario complexity its decisions far exceed purely neural baselines sharing the same backbone ($F1 = 0.803$ against 0.103 on the extreme set), while remaining comparable or worse under pure perceptual degradation such as occlusion; models trained with less expensive supervision produce fewer false positives while recovering the same events. A component ablation separates the symbolic machinery into structural gradient routing from the logic rules and visual grounding from the auxiliary losses. Either alone yields comparable decisions, but only grounding yields interpretable intermediate concepts, and the two come apart under recoloring in both directions: only their combination transfers on the clean recolored set ($F1 = 1.000$ against 0.195 and 0.000), while under combined recoloring and extreme complexity all three degrade and routing alone is best (0.526 against 0.440). Decision accuracy is therefore a poor proxy for concept quality in neuro-symbolic systems.

1 INTRODUCTION

Neural networks trained to detect a complex event receive no prior knowledge about what constitutes that event. Asked to recognize a tool delivery, the network must learn from examples alone that two actors with distinct roles are involved, an instrument changes possession during the transfer from the giver to the receiver, and the two are spatially adjacent at the moment of exchange. Each of these intermediate concepts has to emerge implicitly from the gradient of the final training objective.

When examples are limited or narrowly distributed, the network is not guaranteed to converge on the intended concepts. Geirhos et al. (2020) show that deep networks systematically acquire shortcut decision rules that perform well in-distribution and fail outside it, closely related to the Clever Hans effect documented across machine learning (Lapuschkin et al., 2019; Kauffmann et al., 2025). In the tool handover case, a model may latch onto the motion signature of two hands approaching rather than the underlying notion of possession transfer: it learns what handovers *look like* in the training data, not what a handover *is*. This problem persists even when symbolic reasoning is introduced: a neuro-symbolic model can optimize its decision objective while assigning unintended semantics (*reasoning shortcuts*) to its intermediate concepts (Marconato et al., 2023). Thus, logical consistency and high event-detection accuracy alone do not establish that learned predicates such as role or possession correspond to their intended visual meaning.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The standard remedy is to supervise the intermediate concepts directly. This works, but the cost asymmetry is severe: a clip-level event label is a single binary decision, whereas the intermediate annotation is frame-by-frame, region-by-region labeling of positions, roles and possession states. The asymmetry appears whenever the target event is compositional, defined by a conjunction of perceptual predicates rather than a single pattern.

Abductive learning. Neuro-symbolic AI offers a third option. Instead of requiring the network to discover all concepts from data, domain knowledge enters as differentiable logical rules, constraint losses and domain priors (Dash et al., 2022). When the reasoning engine is differentiable, the loss computed on the cheap final label flows backward through the rules into exactly those perception channels that participated in each derivation. We call this regime *abductive learning*, following Dai et al. (2019), who introduced the term for the same problem: a perception model produces primitive logical facts for which no ground truth exists, constrained only by a final label together with symbolic background knowledge. They bridge the two through logical abduction and consistency optimization; we adopt the term in the same spirit but realize it through a differentiable reasoning engine.

Scope and contributions. Measuring whether a learned intermediate concept is correct requires ground truth for that concept at every frame and location, which real video of compositional events does not provide. We therefore work in a fully controlled synthetic setting, which is what makes the central question answerable at all; a detection-based instantiation on real video is outside the scope of this paper. (i) We build a complete abductive learning pipeline pairing a grid-adapted VideoMamba (Li et al., 2025) backbone with Scallop (Li et al., 2023b), together with a generator for role-based multi-agent video in which actors of different colors transfer tools to each other on a discrete grid, annotated at two cost tiers (Sec. 3–4). (ii) We show the pipeline recovers expensive intermediate concepts from cheap labels alone: at 0% expensive annotation it reaches correlations of 0.965, 0.997 and 0.999 for the two role channels and instrument possession, matching a fully supervised variant; only possession is recovered without an appearance prior, the role channels being anchored by a color prior (Sec. 5.1). (iii) We show this grounding pays off out of distribution, and that lower annotation rates yield higher mean F1 on complex unseen scenes, mainly through fewer false positives (Sec. 5.2). (iv) We decompose the symbolic machinery into two separable functions, structural gradient routing and visual grounding, and show that decision accuracy cannot distinguish them: a model whose intermediate predictions are spatially meaningless matches a fully grounded one on the decision task (Sec. 5.3).

2 Related Work

Injecting symbolic knowledge. Dash et al. (2022) organize knowledge injection by entry point: input representation, loss function, or architecture. Input-level methods transform data before it reaches the network (Lavrač et al., 1991; Liu et al., 2020). Loss-level methods add penalty terms measuring constraint violation (Diligenti et al., 2017; Xu et al., 2018; Hu et al., 2016). Architecture-level methods compile rules into structure, from KBANN (Towell & Shavlik, 1994) to Logic Tensor Networks (Badreddine et al., 2022). Our architecture uses several of these simultaneously and is deliberately not confined to one row of that taxonomy.

Differentiable reasoning engines. Few neuro-symbolic approaches attempt to embed a full symbolic reasoner into the AI pipeline and make it end-to-end differentiable. DeepProbLog(Manhaeve et al., 2021) extends probabilistic logic programming with neural predicates whose distributions come from network outputs; NeurASP (Yang et al., 2020) does the analogous thing for answer set programming. Scallop (Li et al., 2023b) follows the neural-predicate principle on a Datalog-based language and replaces exact probabilistic inference with provenance semirings (Green et al., 2007), which makes its outputs differentiable and lets gradients reach precisely the predicates participating in each proof. In Kautz's taxonomy (Kautz, 2022) this is the tightest integration type; we adopt Scallop for exactly this gradient-routing property.

Weakly supervised neuro-symbolic video understanding. LASER (Huang et al., 2025) learns spatio-temporal scene graphs from video captions by translating captions into temporal logical specifications and differentiably aligning predicted scene graphs with them. Like LASER, we use sym-

bolic temporal structure to route weak supervision into perceptual predicates. The settings differ: LASER targets open-domain scene-graph generation from captions, whereas we study abductive recovery of role and possession concepts from cheap event labels in a controlled multi-agent video domain. This controlled setting lets us evaluate the semantic grounding of every learned intermediate predicate against exact ground truth.

Concept quality and datasets. Neuro-symbolic predictors can achieve high task accuracy while learning intermediate concepts with unintended semanticsMarconato et al. (2023). Their analysis shows that logical constraints, data coverage, optimization objectives, and the concept-extractor architecture can all leave concept semantics underdetermined. While Bortolotti et al. (2024) show that neuro-symbolic models frequently reach high task accuracy while associating the wrong concepts with the input. Our ablation studies this issue in spatio-temporal video: we separate decision-level success from spatially grounded, interpretable intermediate predicates.

Existing synthetic benchmarks target static compositional queries (Johnson et al., 2017), physics-driven video events (Yi et al., 2020; Chen et al., 2022; Li et al., 2023a), spatio-temporal relations between geometric primitives (Agrawal et al., 2023), or interactions among MNIST digits (Apriceno et al., 2021). None of these benchmarks combines temporally structured events with explicit role-based agents whose admissible actions are conditioned on their roles—that is, with the role-governed possession and action transitions required for compositional handover-style events.

3 MODEL

The pipeline has three stages (Appendix Figure 3): a trainable perception module producing structured per-cell predictions, a Scallop reasoner applying domain rules to those predictions, and auxiliary symbolic inductions that shape perception during training only.

Perception module. It maps a clip of T frames to per-cell probability maps of shape (T, H_G, W_G) per event channel. It is built on VideoMamba (Li et al., 2025), so that the same backbone also serves the purely neural baselines, with two modifications for dense prediction. First, the classification head is removed and the per-patch features after the final Mamba block are retained; at 256×256 input and patch size 16 the spatial grid is exactly 16×16, matching the dataset grid by construction. Second, because the selective scan mixes information across all tokens and can dilute the correspondence between a patch feature and its pixel region, heads for temporal events receive $f_{h,w,t} = [\, f^{\text{mamba}}_{h,w,t} \| \text{Adapter}(f^{\text{raw}}_{h,w,t}) \,]$, preserving spatial identity while retaining temporal context. Each event key has a small MLP head emitting one logit per cell, trained with focal loss (Lin et al., 2017).

Symbolic reasoning. The six sigmoid-activated channels (roles `green` and `blue`, possession `actor_holds`, delivery `handover`, and the additional states `green_prepares` and `blue_works`) enter Scallop directly as probabilistic facts; the complete program is given in Appendix A.4. The key intermediate relation, `holds_transfer`, combines role and possession predictions across adjacent cells and consecutive frames. For the directional green-to-blue rule `handover_g2b`, the derivation decomposes as

$$\begin{aligned} p_{\texttt{g2b}}(x,y,t) &= p_{\texttt{handover}}(x,y,t) \cdot p_{\texttt{transfer}}(x,y,t) \\ p_{\texttt{transfer}}(x,y,t) &= f\big(p_{\texttt{green_holds}}(x,y,t-1),\ p_{\texttt{blue_holds}}(x',y',t+1)\big) \\ p_{\texttt{green_holds}}(x,y,t) &= p_{\texttt{holds}}(x,y,t) \cdot p_{\texttt{green}}(x,y,t) \\ p_{\texttt{blue_holds}}(x',y',t) &= p_{\texttt{holds}}(x',y',t) \cdot p_{\texttt{blue}}(x',y',t) \end{aligned} \tag{1}$$

where (x', y') is a neighboring cell and f is the conjunction implemented by the provenance semiring. The primary decision rule likewise requires both a direct neural handover prediction and a verified possession transfer, $\texttt{handover_full}(x,y,t) = \texttt{handover}(x,y,t) \wedge \texttt{holds_transfer}(x,y,t)$. Weaker rules based on perception alone, perception plus spatial evidence, or temporal context alone are weighted more heavily early in training and subsequently annealed away, avoiding the uninformative gradients that the near-zero probability of the full conjunction would yield at initialization. Directional variants additionally permit supervision through

the cheap direction label. Because Scallop's provenance semirings make these derivations differentiable, the loss on the cheap label reaches exactly the channels that participated in each proof (Appendix A.1). Reasoning is restricted to $3 \times 3 \times 7$ local windows placed by non-maximum suppression on agent-detection confidence.

Auxiliary symbolic inductions. The rules specify which predicates must co-occur and in what order, but nothing forces the model to represent them interpretably. Three auxiliary losses close that gap, all active during training only. The *color prior* anchors `green` and `blue` to the known agent appearance; but it is confined to agent identity, says nothing about possession or timing, and in a deployed system would be replaced by any pretrained detector. The *structural constraint loss* encodes scene properties holding regardless of any delivery: spatial uniqueness (minimizing the entropy of each agent channel's spatial distribution), existence (so uniqueness cannot be satisfied by predicting nothing), mutual exclusivity, holds co-location and spatial concentration of handover predictions, plus cheap-label-gated terms for handover proximity and holds transition. The *visual consistency loss* stabilizes `actor_holds` outside handover sequences, where it otherwise drifts: detached EMA prototypes μ^+ (confidently held) and μ^- (present but not holding) are collected during handover sequences, and elsewhere each active-agent cell is scored by $s_{\text{rel}} = \cos(f, \mu^+) - \cos(f, \mu^-)$, a push term suppressing possession where the feature resembles a non-holder and a boost term raising it where it resembles a holder. Gradient flow across the overlapping losses is controlled by explicit stop-gradients (Appendix A.1).

Supervision control. The composite objective is $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{scallop}} + \lambda_{\text{str.}}\,\mathcal{L}_{\text{structural}} + \lambda_{\text{col.}}\,\mathcal{L}_{\text{color}} + \lambda_{\text{vis.}}\,\mathcal{L}_{\text{visual}} + \lambda_{\text{prec.}}\,\mathcal{L}_{\text{perception}}$, where $\mathcal{L}_{\text{perception}}$ is the direct cell-level loss, active only when expensive annotations are available. An expensive data rate $r_{\text{exp}} \in [0, 1]$ controls this: annotations are supplied iff $r_{\text{exp}} > 0$ and $\text{step} \bmod \lfloor 1/r_{\text{exp}} \rfloor = 0$, keeping exposure constant in every epoch rather than accumulating as it would under random per-sample sampling. At $r_{\text{exp}} = 0$ the only task-specific supervision is the cheap decision label flowing through the Scallop rules. Because every symbolic component operates independently of the expensive annotations, removing them degrades the objective gracefully rather than breaking training. The complete optimizer configuration, learning-rate schedule, and weights of all loss terms are reported in Appendix A.5, Table 4.

4 Experimental Setup

Data. All experiments use a purpose-built synthetic generator, which is what makes the learned intermediate concepts measurable: it supplies ground truth for every concept at every frame and cell. In a top-down 2D scene, a green agent collects an object, moves to an available blue agent and delivers the object "handover"; the blue agent works with it and eventually returns it. Role-specific state machines branch, so not every pickup ends in a handover (Appendix Figure 4). The event is defined temporally rather than visually, $\text{handover}(a_1, a_2, o) \equiv \text{holds}(a_1, o) \wedge X\,\text{holds}(a_2, o) \wedge a_1 \neq a_2$ with X the next-state operator, so it cannot be read off a single frame, and holding states persist across many consecutive frames, so the mere presence of a `holds` predicate is uninformative. Two ground-truth tiers are used: cheap frame-level event flags (the handover flag and its direction) and expensive per-frame cell coordinates and states for every agent, equivalent in a real setting to frame-level bounding-box or key-point labeling. Model selection used the reasoning loss on a held-out in-distribution validation split (10% of the clean training videos, partitioned by video rather than by clip); no out-of distribution test set was used for checkpoint selection or early stopping.

Training protocol. Every model, neuro-symbolic and baseline alike, is trained *exclusively* on the simplest configuration: exactly two agents, clean handovers, no occlusion, a single fixed instrument appearance. No difficult or degraded scenario is ever seen during training; all runs use a weighted sampler for a balanced handover distribution. The neuro-symbolic models are trained with three seeds, the baselines and ablation variants with the first seed only. Evaluation then happens entirely out of distribution, mirroring a common deployment constraint in sensitive domains where training data is collected under controlled laboratory conditions while deployment must survive real variability.

Test conditions. Test sets fall into multiple tiers: Perceptual degradation intermittently occludes the instrument (`Occl.`), hides it (`h30`), or randomizes its shape and color on every pickup (`Irand`),

Table 1: Perception quality (CC) by expensive annotation rate, on the training distribution (Clean) and the hardest recolored out-of-distribution set (Extr.+col). Higher is better. The three core channels are near-saturated at 0%; only `handover` materially benefits from direct supervision. Full results over all test sets are in Appendix Table 6.

	Clean				Extr.+col			
Channel	0%	5%	50%	100%	0%	5%	50%	100%
Green	0.965	0.930	0.977	0.971	0.789	0.767	0.752	0.720
Blue	0.997	0.998	0.931	0.887	0.687	0.807	0.703	0.663
Actor holds	0.999	0.999	0.864	0.981	0.783	0.876	0.792	0.912
Handover	0.093	0.198	0.903	0.997	0.224	0.438	0.742	0.890

testing whether the model recognizes possession rather than a fixed visual template. Semantic complexity adds two additional agents creating pairing ambiguity (`4a`), failed transfers and approach-only sequences which look like a handover (following a handover trajectory) but with an empty-handed agent (`FO`) and the most complex scenario with additional intra-role green-to-green fake handovers (`Extr.`). Color transfer replaces the trained green/blue palette with unseen orange/red, alone (`Colors`) and on top of the extreme condition (`Extr.+col`).

Metrics. At the *perception level* we report the Pearson correlation coefficient (CC) between predicted probability map and ground-truth grid, which captures relative spatial structure but is invariant to absolute confidence; IoU@0.5, which binarizes first and is therefore sensitive to confidence; and normalized mean error (NME), a graded measure distinguishing near-misses from far-misses. Only the neuro-symbolic models have an explicit grid representation, so this level does not apply to the baselines. At the *decision level* we report F1 under peak-based event evaluation (Katsarou et al., 2026): per-frame confidences are smoothed with a Gaussian kernel ($\sigma = 3$ frames) into a temporal signal, events are read off as peaks above [height/prominence $= 0.05/0.025$], and a predicted peak counts as correct if it lies within four frames of a ground-truth event. These parameters are fixed across all models and test sets and were set on the in-distribution validation split. Peak-based evaluation replaces per-frame thresholding with this single shared configuration, making the comparison robust to differences in calibration and to the temporal offset between the baselines (which predict the center frame) and our model (which predicts every frame).

Baselines. **VM-bin** and **VM-dir** are standard VideoMamba sequence classifiers trained end-to-end on the same data with the binary handover label and the three-class direction label respectively, so VM-dir receives exactly the directional information our model gets as cheap supervision. **VM-grid** uses the same grid-decomposed backbone adapted to the neuro-symbolic architecture but predicts handovers directly from the perception grid with no Scallop reasoning, isolating any advantage conferred by grid decomposition itself in comparison to the original vanilla VideoMamba.

5 Results

5.1 Intermediate concepts are learned without expensive data

Table 1 reports perception quality across annotation rates, with further metrics in Appendix Figure 5 and Tables 7 and 8. For the three core channels, the 0% model reaches CC $= 0.965$ for `green`, 0.997 for `blue` and 0.999 for `actor_holds` on clean data, matching or exceeding the fully supervised variant (0.971, 0.887, 0.981). The role channels are anchored by the color prior, but `actor_holds` receives no spatial signal other than the cheap decision label propagated through the Scallop rules and the structural constraints: possession is localized abductively, without any grid-level annotation. Correct spatial structure does not imply calibrated confidence, however. Because CC is invariant to absolute confidence, it cannot reveal that the 0% `actor_holds` map is under-confident: its IoU@0.5 is only 0.539, against 0.966 at 100%. A small amount of expensive data largely closes this gap (0.915 at 5%), so abduction supplies the structure of the concept while sparse direct supervision mainly sharpens its confidence.

Table 2: Handover detection F1 across test conditions of increasing difficulty. NeSym rows report the mean F1 over three training seeds, with the sample standard deviation shown below in smaller print; VM baselines are single-run results. Bold marks the best mean per column (ties are all bold). Neuro-symbolic models are evaluated via the `holds_transfer` rule output (Sec. 3); 0.000 means no valid detection was produced. The last two columns replace green/blue agents with orange/red. Precision and recall are in Appendix Table 9.

	Scenario complexity					Perturbations			
Model	Clean	Irand	4a	4a+FO	Extr.	Occl.	h30	Colors	Extr.+col
NeSym 0%	0.982 ±0.030	0.951 ±0.050	**0.955** ±0.078	0.708 ±0.253	0.803 ±0.166	0.810 ±0.017	0.838 ±0.021	**0.815** ±0.321	**0.464** ±0.125
NeSym 5%	0.963 ±0.032	0.959 ±0.036	0.918 ±0.091	**0.743** ±0.298	**0.812** ±0.158	0.712 ±0.066	**0.858** ±0.016	0.486 ±0.214	0.280 ±0.214
NeSym 50%	**1.000** ±0.000	**0.977** ±0.010	0.882 ±0.025	0.601 ±0.149	0.667 ±0.026	0.882 ±0.045	0.775 ±0.148	0.678 ±0.162	0.416 ±0.123
NeSym 100%	**1.000** ±0.000	0.965 ±0.011	0.868 ±0.215	0.477 ±0.160	0.738 ±0.055	0.872 ±0.131	0.821 ±0.038	0.649 ±0.081	0.419 ±0.116
VM-bin	0.962	0.519	0.842	0.353	0.103	0.974	0.806	0.000	0.000
VM-dir	**1.000**	0.827	0.160	0.158	0.231	**1.000**	0.640	0.026	0.232
VM-grid	0.602	0.442	0.556	0.140	0.263	0.545	0.482	0.400	0.424

`handover` is the exception, rising from CC $= 0.093$ at 0% to 0.997 at 100%. Its ground truth spans several adjacent cells during the transfer, and without direct supervision nothing resolves that ambiguity into a sharp map. The prediction is nonetheless not random: NME stays low at 0% (0.056 on clean data against 0.008 at 50%, Appendix Table 7), while no cell exceeds 0.5 (IoU@0.5 $= 0.000$, Table 8). The map is centered near the right place but spread across neighbors at low confidence, precisely the combination that yields low CC and low NME at once. Because decisions are read from `holds_transfer`, which does not depend on this channel, its weakness does not limit detection performance (Section 5.2).

5.2 Concept grounding pays off on unseen data complexity

Table 2 reports decision performance (NeSym: mean over three seeds), plotted in Appendix Figures 7 and 8. On clean and near-distribution data every model except VM-grid performs well, but the separation opens on multi-agent data and widens with complexity. On the extreme set the end-to-end baselines collapse (F1 ≈ 0.10-0.26) while the 0% and 5% neuro-symbolic models average above 0.80 (with a seed std of ≈ 0.16). The rules act as an explicit structural filter: a detection requires a verifiable `holds_transfer` between two differently-roled agents across consecutive frames, and that compositional requirement is hard to approximate by pattern matching on full-frame features. VM-grid indicates that switching to the grid-decomposed backbone without the additional symbolic reasoning layer performs worse than both baselines.

Where the baselines win, and where they fail. On occlusion and hidden-instrument sets the baselines remain competitive and sometimes better (VM-dir reaches 1.000 on occlusion against 0.810). These perturbations leave scene structure intact, so learned temporal patterns still apply, whereas `holds_transfer` might miss a delivery when the instrument is not visible in every frame. This is a deliberate trade-off: the explicit visibility requirement costs accuracy here and is exactly the behavior wanted on failed and approach-only sequences. Color transfer inverts the picture: the neural baselines essentially fail (VM-bin produces no detections on either color set), while the 0% neuro-symbolic model reaches a mean F1 of 0.815 ± 0.321 on the clean color-transfer set (1.000 for two of three seeds). The prior does not tie the model to the training palette: its role channels still localize the recolored agents (CC 0.83/0.78, Appendix Table 6), and the ablation (Section 5.3) shows that this anchoring is part of what makes the transfer concept survive recoloring.

Less supervision, more robustness. On the complexity sets the annotation rates split into two regimes: in the mean the low-rate models (0%, 5%) outperform the high-rate models (50%, 100%),

Table 3: Symbolic induction ablation at 0% expensive data. Bold marks the best F1 per dataset. Both single-function variants approach the full model on the decision task, yet each collapses on color transfer where the full model does not. Remaining datasets in Appendix Table 10.

Dataset	Full induction		Scallop only		VC transfer		VC neural
	F1	(P/R)	F1	(P/R)	F1	(P/R)	F1
Clean	**1.000**	(1.00/1.00)	**1.000**	(1.00/1.00)	**1.000**	(1.00/1.00)	**1.000**
Occluded	0.829	(0.94/0.74)	0.650	(0.48/1.00)	0.696	(0.60/0.82)	**0.886**
Irand	**0.988**	(0.98/1.00)	0.938	(0.93/0.95)	0.857	(0.82/0.90)	0.976
4-actor	**1.000**	(1.00/1.00)	**1.000**	(1.00/1.00)	0.852	(1.00/0.74)	0.906
4a+FO50	0.902	(0.82/1.00)	0.905	(1.00/0.83)	**0.939**	(0.89/1.00)	0.585
4a+HARD	0.909	(0.87/0.95)	0.829	(0.85/0.81)	**0.930**	(0.91/0.95)	0.667
Extreme	0.815	(0.72/0.94)	**0.825**	(0.93/0.74)	0.765	(0.67/0.89)	0.594
Colors	**1.000**	(1.00/1.00)	0.195	(1.00/0.11)	0.000	(0.00/0.00)	0.763
Extr.+col	0.440	(0.52/0.38)	**0.526**	(0.54/0.52)	0.000	(0.00/0.00)	0.217

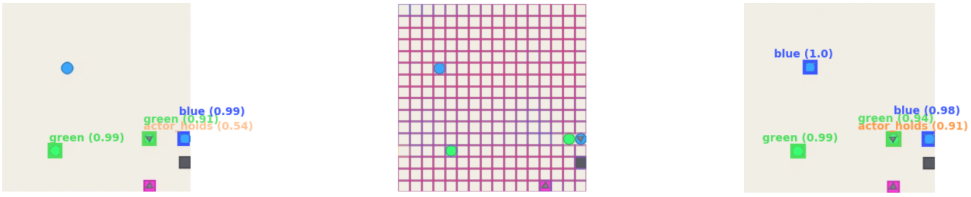
with F1 0.803–0.812 against 0.667–0.738 on the extreme set and 0.708–0.743 against 0.477–0.601 On 4a+FO the gain is entirely in precision: recall is near-perfect at every rate while precision falls from 0.59–0.65 to 0.33–0.44 (Appendix Table 9); on the extreme set precision and recall contribute about equally. With direct grid supervision the perception module reproduces the specific spatial patterns of the training data, which produce spurious activations under more agents or different layouts; Appendix Figure 9 of seed 1 shows a 100% model whose spurious `blue` activation triggers `holds_transfer`. The low-rate models are anchored to no such patterns: they learn *what information the rules need* rather than *what the training data looks like*. Seed variance is large on the hard sets (up to ±0.30), so the regime gap holds in the mean but not seed by seed, and on the perturbation and color sets no consistent ordering emerges.

5.3 Ablation: two separable functions of symbolic knowledge

To show how much each component contribute in our neuro-symbolic architecture, we train three variants at 0% expensive data, identical in backbone, grid decomposition and windowing, differing only in which signals guide perception; each is a single first-seed run, Full induction being the first seed of NeSym 0%. **Full induction** uses Scallop rules plus all auxiliary losses (color prior CP, structural constraints SC, visual consistency VC): both routing and grounding active. **Scallop only** removes all auxiliary perception losses, leaving the Scallop decision loss and its backward pass as the sole training signal; the rules define *how* gradients propagate to each channel, but nothing constrains *what* those channels should represent. **Visual constraints** removes the Scallop module entirely during training, leaving only CP, SC and VC; at inference we evaluate it twice, reading the neural handover channel directly (*VC neural*) and attaching the `holds_transfer` rule post hoc (*VC transfer*), the latter having never seen a Scallop gradient.

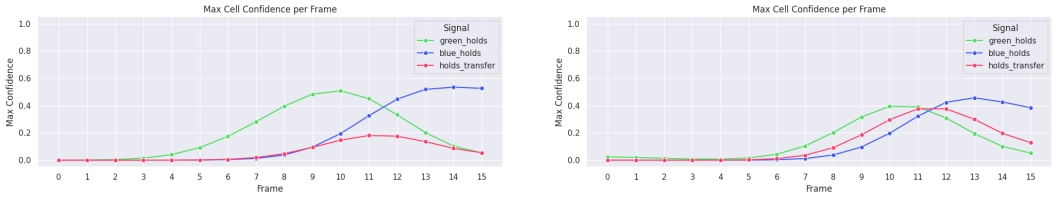
All three configurations reach comparable F1 on standard and moderately complex data (Table 3, Appendix Figure 10). All reach 1.000 on clean data. On the extreme set, full induction gets 0.815, Scallop only 0.825, and the visual-constraints model 0.765 once the transfer rule is attached post hoc. Gradient routing alone and visual grounding alone are each *sufficient* for strong handover detection, including far out of distribution. The two do differ in operating characteristics: full induction runs high-recall (0.94 on extreme) at moderate precision (0.72), while Scallop only inverts this (0.93 precision, 0.74 recall). The auxiliary losses improve recall by spatially grounding the channels, which helps the rules fire under noisy conditions; without that grounding the model is conservative, firing only when its self-organized representations produce a high-confidence rule match.

Intermediate representations diverge sharply. Decision parity does not survive inspection of what the models actually predict (Figure 1). Full induction and visual constraints localize `green`, `blue` and `actor_holds` at the correct cells. The Scallop-only model's activations are spatially diffuse and carry no per-frame interpretable meaning at all, despite equal decision accuracy. This is consistent with the reasoning-shortcut phenomenon of Marconato et al. (2023): satisfying a symbolic decision objective does not by itself identify the intended semantics of latent predicates. Here,



(a) Full Induction (b) Scallop Only (c) Visual Constraints

Figure 1: Intermediate cell predictions for the same frame under each ablation configuration. Full induction (left) and visual constraints (right) produce spatially precise predictions matching ground truth; Scallop-only (center) is spatially diffuse with no discernible per-frame pattern.



(a) Temporal activations of the Full Induction model. (b) Temporal activations of the Scallop Only model.

Figure 2: Temporal traces of `green_holds`, and `blue_holds` confidences, `holds_transfer` rule activation for the full induction (left) and the Scallop-only (right). Despite spatial incoherence, the Scallop-only preserves the green-to-blue transition around the handover frame.

the Scallop-only model learns temporally useful internal channels, but auxiliary grounding is required to make those channels correspond to verifiable spatial concepts. Although the per-frame spatial distribution looks arbitrary, the maximum-confidence `green_holds` and `blue_holds` cells follow a clean transition around the handover, green falling as blue rises (Figure 2). *The rules shape when channels activate, not where.*

Two distinct symbolic functions. The first is **structural gradient routing**: during training the rules define the computational graph through which the decision loss reaches perception. Following the decomposition in Eq. 1, $\partial\mathcal{L}/\partial\theta_{\text{perception}}$ reaches all four channels, so every channel receives decision-loss gradient and no shortcut can bypass one. Crucially, this holds *regardless of interpretability*: the rules dictate which channels participate and when they fire, not where in the grid. The second function is **visual and semantic grounding**, supplied by the auxiliary losses rather than the rules, which define what the symbols are supposed to *mean*. Without them `actor_holds` is an abstract pattern that happens to satisfy the temporal rules; with them it becomes a spatially precise possession prediction an observer can verify against the scene.

Attributing the `holds_transfer` output back onto perception (Appendix Figure 6) shows the same divergence. The routing is identical in both models, yet for full induction the cell-level attribution concentrates on the exact agent cells, so the decision can be traced step by step, whereas for Scallop only the diffuse underlying predictions smear it across the grid. At the *pixel* level the contrast largely disappears: both attend broadly to the agents, so both extract comparable visual information but encode it differently, into precise cell predictions or into a functionally equivalent but opaque distribution, which plausibly explains their comparable decisions. One difference remains telling: among the pixels driving `actor_holds`, full induction shows markedly higher saliency where the instrument is actually held. It responds to the holder directly and catches more atypical handovers (higher recall), whereas Scallop only reads holding from overall scene composition and commits only when the whole configuration fits (higher precision).

Where the combination is necessary. Decision parity between the two functions holds on most test sets but breaks on color transfer, and it breaks in *both* directions. On the recolored clean set, Scallop only drops to F1 $= 0.195$ and visual constraints with the post-hoc transfer rule to 0.000, while full induction reaches 1.000. Routing without grounding leaves the channels with no stable

perceptual anchor when appearance changes. Grounding without routing leaves the role channels tied to the training palette, with no rule-derived pressure to represent the relational structure that survives recoloring: the transfer rule attached post hoc never fires, even though the neural handover channel of the same model still reaches 0.763. The CP+Scallop variant (Appendix Table 10) locates much of the gap: adding only the color prior to the rules raises Colors F1 from 0.195 to 0.725. Anchoring the role channels to agents, rather than leaving them self-organized, is therefore itself a source of robustness to appearance shift, and the structural and visual-consistency losses close the remaining gap. This benefit is limited to appearance shift: under combined recoloring and extreme complexity all variants degrade, and the prior brings no gain (0.468 against 0.526 for Scallop only, 0.440 for full induction). Either function alone is enough for a benchmark number, but on a shift in agent appearance only their combination transfers.

6 Discussion

Learning the event concepts yields robustness on unseen scenarios. Forcing the model to represent the concepts underlying the event, the actor roles and states that define a handover, produces a model that holds up far better out of distribution. A purely neural detector sharing the same backbone learns whatever appearance regularities separate handovers from non-handovers in the training data, and those regularities break as soon as the scene changes: more agents, recolored agents, and fake or failed transfers all collapse it. Our model degrades gracefully because its decision is routed through explicit intermediate predicates, committing it to *what an instrument delivery is*, a possession transfer between two differently-roled agents across consecutive frames, rather than to what handovers looked like during training. The clearest evidence is that the models given the least supervision generalize best. Routing through predicates alone, however, does not guarantee that they acquire their intended visual semantics: the Scallop-only model reaches comparable decision performance while its intermediate channels remain spatially diffuse, a reasoning shortcut in the sense of Marconato et al. (2023).

Structural routing and visual grounding. The Scallop rules define which perception channels participate in each derivation and in what temporal order, while the auxiliary losses tie those channels to spatially meaningful predictions. Routing alone already suffices for a correct decision, even far out of distribution; on in-distribution and complex data, what grounding adds is not accuracy but meaning; under appearance shift it also adds robustness. When one function is worth trading against the other is a question our experiments open but do not settle.

Limitations. The color prior is the one place supervision enters outside the cheap label: confined to agent identity and replaceable by a pretrained detector, it nonetheless means 0% expensive data is not 0% prior knowledge. Without it and the other auxiliary losses the role channels are not grounded (Scallop only, Sec. 5.3) and it contributes substantially to robustness under recoloring. so its effect cannot be separated from that of the rules. Reasoning is clip-local (16-frame clips), which keeps the baseline comparison fair but leaves sequence-level symbolic reasoning unexplored, and the `handover` channel is the one concept abduction does not sharpen, its ground truth being spatially ambiguous.

7 Conclusion

We built an abductive learning pipeline requiring no expensive intermediate annotation and showed, against exact ground truth, that it works: coupling a grid-adapted VideoMamba backbone to Scallop recovers agent roles and instrument possession at correlations above 0.95 from cheap event labels alone, with large out-of-distribution gains over neural baselines sharing the same backbone. Less expensive supervision yields better generalization, because the model learns what the rules require rather than what the training data looks like. The ablation separates the symbolic contribution into structural gradient routing and visual grounding: either alone produces a strong decision model, only grounding produces interpretable concepts, and only their combination survives a change in agent appearance. Concept quality therefore has to be measured directly; the decision metric will not reveal it.

Reproducibility statement

The code for our models, synthetic-data generator, experiment configurations, complete Scallop program, and training hyperparameters will be shared at [Git-URL to be added in the camera-ready]. The full symbolic rules are additionally included in Appendix A.4; optimization settings and loss weights are reported in Appendix A.5, Table 4.

AI use statement

In this work, we used generative AI tools (Claude) for; edit a research paper to improve readability, summarize existing literature, and edit code. We have reviewed all AI-assisted writing, and verified and tested LLM-generated code for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Palaash Agrawal, Haidi Azaman, and Cheston Tan. STUPD: A synthetic dataset for spatial and temporal relation reasoning. *arXiv preprint arXiv:2309.06680*, 2023.

Gianluca Apriceno, Andrea Passerini, and Luciano Serafini. A neuro-symbolic approach to structured event recognition. In *28th International Symposium on Temporal Representation and Reasoning (TIME 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic tensor networks. *Artificial Intelligence*, 303:103649, 2022. doi: 10.1016/j.artint.2021.103649.

Samuele Bortolotti, Emanuele Marconato, Tommaso Carraro, Paolo Morettin, Emile van Krieken, Antonio Vergari, Stefano Teso, and Andrea Passerini. A neuro-symbolic benchmark suite for concept quality and reasoning shortcuts. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B. Tenenbaum, and Chuang Gan. ComPhy: Compositional physical reasoning of objects and events from videos. *arXiv preprint arXiv:2205.01089*, 2022.

Wang-Zhou Dai, Qiuling Xu, Yang Yu, and Zhi-Hua Zhou. Bridging machine learning and logical reasoning by abductive learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Tirtharaj Dash, Sharad Chitlangia, Aditya Ahuja, and Ashwin Srinivasan. A review of some techniques for inclusion of domain-knowledge into deep neural networks. *Scientific Reports*, 12(1): 1040, 2022. doi: 10.1038/s41598-021-04590-0.

Michelangelo Diligenti, Marco Gori, and Claudio Saccà. Semantic-based regularization for learning and inference. *Artificial Intelligence*, 244:143–165, 2017. doi: 10.1016/j.artint.2015.08.011.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. doi: 10.1038/s42256-020-00257-z.

Todd J. Green, Grigoris Karvounarakis, and Val Tannen. Provenance semirings. In *Proceedings of the Twenty-Sixth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 31–40, 2007. doi: 10.1145/1265530.1265535.

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. Harnessing deep neural networks with logic rules. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2410–2420, 2016. doi: 10.18653/v1/P16-1228.

Jiani Huang, Ziyang Li, Mayur Naik, and Ser-Nam Lim. Laser: A neuro-symbolic framework for learning spatio-temporal scene graphs with weak supervision. In *International Conference on Learning Representations*, volume 2025, pp. 41977–41999, 2025.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1988–1997, 2017. doi: 10.1109/CVPR.2017.215.

Katerina Katsarou, George Zountsas, Karam Tomotaki-Dawoud, Alexander Ehrenhoefer, Paul Chojecki, David Przewozny, Igor Maximilian Sauer, Amira Mouakher, and Sebastian Bosse. Event-level detection of surgical instrument handovers in videos with interpretable vision models. *arXiv preprint arXiv:2604.07577*, 2026.

Jacob Kauffmann, Jonas Dippel, Lukas Ruff, Wojciech Samek, Klaus-Robert Müller, and Grégoire Montavon. Explainable AI reveals clever hans effects in unsupervised learning models. *Nature Machine Intelligence*, 7(3):412–422, 2025. doi: 10.1038/s42256-025-01000-2.

Henry A. Kautz. The third AI summer: AAAI robert s. engelmore memorial lecture. *AI Magazine*, 43(1):105–125, 2022. doi: 10.1002/aaai.12036.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature Communications*, 10(1):1096, 2019. doi: 10.1038/s41467-019-08987-4.

Nada Lavrač, Sašo Džeroski, and Marko Grobelnik. Learning nonrecursive definitions of relations with LINUS. In *Machine Learning — EWSL-91*, volume 482 of *Lecture Notes in Computer Science*, pp. 265–281. Springer, 1991. doi: 10.1007/BFb0017020.

Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. VideoMamba: State space model for efficient video understanding. In *Computer Vision – ECCV 2024*, pp. 237–255. Springer Nature Switzerland, 2025. ISBN 978-3-031-73347-5.

Zhuowan Li, Xingrui Wong, Elias Stengel-Eskin, Adam Kortylewski, Wufei Ma, Benjamin Van Durme, and Alan Yuille. Super-CLEVR: A virtual benchmark to diagnose domain robustness in visual reasoning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14963–14973, 2023a. doi: 10.1109/CVPR52729.2023.01437.

Ziyang Li, Jiani Huang, and Mayur Naik. Scallop: A language for neurosymbolic programming. *Proceedings of the ACM on Programming Languages*, 7(PLDI):1463–1487, 2023b. doi: 10.1145/3591280.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2999–3007, 2017. doi: 10.1109/ICCV.2017.324.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. K-BERT: Enabling language representation with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 2901–2908, 2020.

Robin Manhaeve, Sebastijan Dumančić, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. Neural probabilistic logic programming in DeepProbLog. *Artificial Intelligence*, 298:103504, 2021. doi: 10.1016/j.artint.2021.103504.

Emanuele Marconato, Stefano Teso, Antonio Vergari, and Andrea Passerini. Not all neuro-symbolic concepts are created equal: Analysis and mitigation of reasoning shortcuts. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 72507–72539. Curran Associates, Inc., 2023. doi: 10.52202/075280-3170.

Geoffrey G. Towell and Jude W. Shavlik. Knowledge-based artificial neural networks. *Artificial Intelligence*, 70(1–2):119–165, 1994. doi: 10.1016/0004-3702(94)90105-8.

Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, and Guy Van den Broeck. A semantic loss function for deep learning with symbolic knowledge. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5502–5511, 2018.

Zhun Yang, Adam Ishay, and Joohyung Lee. NeurASP: Embracing neural networks into answer set programming. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1755–1762, 2020.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. CLEVRER: Collision events for video representation and reasoning. In *International Conference on Learning Representations (ICLR)*, 2020.

A Appendix

A.1 Gradient Flow Design

With several losses acting on overlapping parameters, uncontrolled propagation creates circular dependencies. Holds co-location, for instance, could be minimized either by moving possession toward an agent or by moving the agent toward possession; only the former is intended, since agent positions are separately anchored by the color prior. Explicit detachment resolves this:

$$\mathcal{L}_{\text{coloc}} = \text{mean}(\text{ReLU}(p_{\text{holds}} - \text{dilate}(\text{sg}[p_{\text{actor}}]))) \,, \tag{2}$$

where $\text{sg}[\cdot]$ is the stop-gradient operator. This pattern recurs throughout the auxiliary losses. The general principle: channels anchored by stronger inductive signals serve as fixed spatial references for channels learned more indirectly.

A.2 Architecture Overview

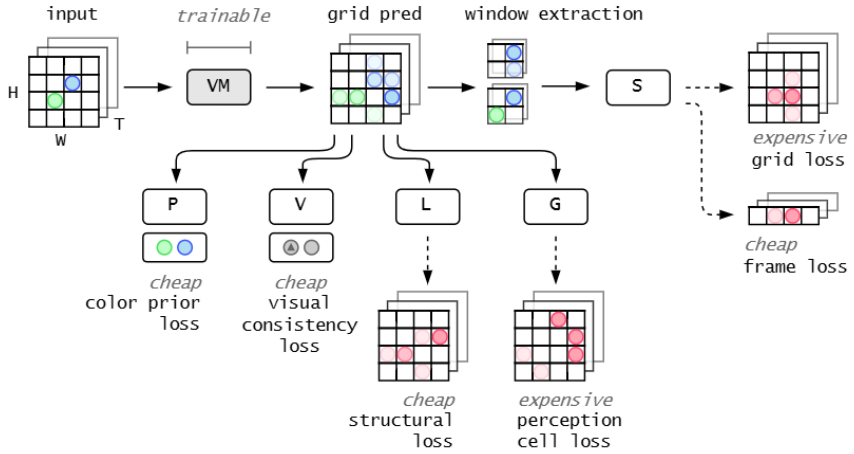
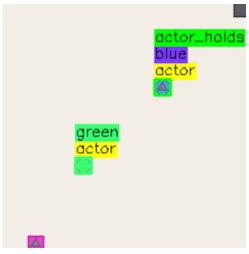
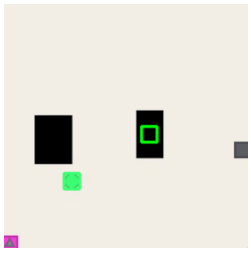


Figure 3: Architecture overview. VM is the VideoMamba backbone, P a prior over agent appearance, L a likelihood-style penalty on implausible predictions, G a cell-level loss active only when expensive annotations are available, and S the Scallop reasoner, which computes a loss on either the cell level (expensive) or the frame level (cheap).

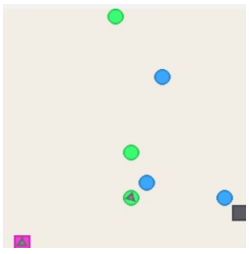
A.3 Example Scenes



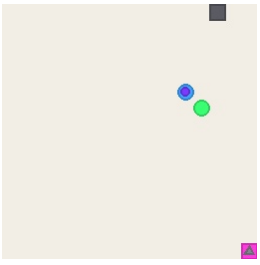
(a) Few actors, no occlusions (bounding boxes and labels visualized)



(b) Few actors with occlusions (bounding boxes visualized)



(c) Many actors in a complex scene



(d) Actor holding a randomized instrument

Figure 4: Example generated scenes at varying complexity: few agents without occlusion, few agents with occlusion (bounding boxes visualized), a dense multi-agent scene, and an agent holding a randomized instrument.

A.4 COMPLETE SCALLOP RULES

Listing 1 contains the complete Scallop program used for the synthetic-data experiments. It defines the discrete spatial and temporal domains, safe temporal and neighborhood lookups, role-specific possession relations, transfer patterns, and the final handover predicates.

Listing 1: Scallop rules for the synthetic data.

```
// === CONSTANT DOMAINS ===
rel TIME_RANGE = {0, 1, 2, 3, 4, 5, 6, 7}
rel GRID_RANGE = {0, 1, 2}
rel neighbor_offset = {
   (-1, -1), (-1, 0), (-1, 1),
   ( 0, -1), ( 0, 1),
   ( 1, -1), ( 1, 0), ( 1, 1)
}

// === SAFE ARITHMETIC LOOKUPS ===
type valid_neighbor(x: i32, y: i32, nx: i32, ny: i32)
type time_prev1(t: i32, tp: i32)
type time_prev2(t: i32, tp: i32)
type time_next1(t: i32, tn: i32)
type time_next2(t: i32, tn: i32)

// === ROLE IDENTIFICATION ===
rel is_green(x, y, t) = green(x, y, t)
rel is_blue(x, y, t) = blue(x, y, t)

rel neighbor_is_different(x, y, nx, ny, t) =
   TIME_RANGE(t) and is_green(x, y, t) and is_blue(nx, ny, t)
rel neighbor_is_different(x, y, nx, ny, t) =
   TIME_RANGE(t) and is_blue(x, y, t) and is_green(nx, ny, t)

// === TEMPORAL EVIDENCE ===
rel held_before(x, y, t) = time_prev1(t, tp) and holds(x, y, tp)

rel neighbor_with_person(x, y, t) =
   TIME_RANGE(t) and
   valid_neighbor(x, y, nx, ny) and
   (green(nx, ny, t) or blue(nx, ny, t))

rel neighbour_holds_after(x, y, t) =
   TIME_RANGE(t) and
   valid_neighbor(x, y, nx, ny) and
   time_next1(t, tn) and holds(nx, ny, tn)

rel neighbour_holds_after(x, y, t) =
   TIME_RANGE(t) and
   valid_neighbor(x, y, nx, ny) and
   neighbor_is_different(x, y, nx, ny, t) and
   time_next2(t, tn) and holds(nx, ny, tn)

// === HOLDS RULES ===
rel green_holds(x, y, t) = holds(x, y, t) and green(x, y, t)
rel blue_holds(x, y, t) = holds(x, y, t) and blue(x, y, t)

rel holds_transfer_green_to_blue(x, y, t) =
   valid_neighbor(x, y, nx, ny) and
   time_prev1(t, tp) and green_holds(x, y, tp) and
   time_next1(t, tn) and blue_holds(nx, ny, tn)

rel holds_transfer_blue_to_green(x, y, t) =
   valid_neighbor(x, y, nx, ny) and
   time_prev1(t, tp) and blue_holds(x, y, tp) and
   time_next1(t, tn) and green_holds(nx, ny, tn)

rel holds_transfer(x, y, t) =
   holds_transfer_green_to_blue(x, y, t)

rel holds_transfer(x, y, t) =
   holds_transfer_blue_to_green(x, y, t)

// === HANDOVER PATTERNS ===
rel handover_neural(x, y, t) =
   GRID_RANGE(x) and GRID_RANGE(y) and TIME_RANGE(t) and
   handover(x, y, t)

rel handover_spatial(x, y, t) =
   handover(x, y, t) and
```

```
    GRID_RANGE(x) and GRID_RANGE(y) and TIME_RANGE(t) and
    neighbor_with_person(x, y, t)

rel handover_context(x, y, t) =
    GRID_RANGE(x) and GRID_RANGE(y) and TIME_RANGE(t) and
    held_before(x, y, t) and
    neighbour_holds_after(x, y, t)

// === FULL HANDOVER ===
rel handover_full(x, y, t) =
    GRID_RANGE(x) and GRID_RANGE(y) and TIME_RANGE(t) and
    handover(x, y, t) and
    holds_transfer(x, y, t)

// === DIRECTIONAL HANDOVER ===
rel handover_g2b(x, y, t) =
    GRID_RANGE(x) and GRID_RANGE(y) and TIME_RANGE(t) and
    handover(x, y, t) and holds_transfer_green_to_blue(x, y, t)

rel handover_b2g(x, y, t) =
    GRID_RANGE(x) and GRID_RANGE(y) and TIME_RANGE(t) and
    handover(x, y, t) and holds_transfer_blue_to_green(x, y, t)
```

A.5 Training Configuration

Table 4 reports the optimization settings, learning-rate schedule, and relative weights of the training objectives used in all experiments.

Table 4: Training hyperparameters, scheduling, and loss weights.

Parameter	**Value**
Optimizer	
Perception LR	1×10^{-4}
Scallop LR	1×10^{-4}
Weight decay	1×10^{-4}
Gradient clipping	1.0
Warmup batches	0
Scheduler	
Type	ReduceLROnPlateau
Patience	5 epochs
Factor	0.5
Monitor	`val/decision_loss`
Loss weights	
Perception	20.0
Penalty	0.0
Structural constraint	0.5
Color prior	1.0
Visual consistency	100.0
General	
Precision	16-mixed (AMP)
Seed	[10, 50, 100]
Num workers	8
Save top-k	3

A.6 Gradient Flow per Loss Component

Table 5 summarizes the intended gradient flow for every training objective. In particular, detachment prevents losses that supervise possession or action states from changing the role channels used as spatial references.

Table 5: Gradient flow design for each loss component. "Receives gradient" indicates which perception channels are updated by each loss. "Detached" indicates channels used as references without gradient flow.

Loss component	**Receives gradient**	**Detached**
Scallop rules	All channels involved in the fired rules (automatic via provenance)	—
Structural: uniqueness	`green`, `blue`	—
Structural: exclusivity	`green`, `blue`	—
Structural: holds co-location	`actor_holds`	`green`, `blue`
Structural: holds transition	`actor_holds`	`green`, `blue`
Structural: handover proximity	`green`, `blue`	—
Structural: prepares co-location	`green_prepares`	`green`
Structural: works co-location	`blue_works`	`blue`
Color prior	`green`, `blue`	—
Visual consistency	`actor_holds`	Features (prototype)

A.7 Additional Perception Metrics

Table 6: Perception quality (CC) across expensive annotation rates and all test datasets. Higher is better; 1.0 is perfect correlation with ground truth.

Channel	Rate	Clean	Irand	4a+FO50	4a+FO50+ir	Colors	Extr.+col
Green	0%	0.965	0.956	0.949	0.933	0.828	0.789
	5%	0.930	0.933	0.935	0.928	0.790	0.767
	50%	0.977	0.979	0.975	0.973	0.795	0.752
	100%	0.971	0.905	0.928	0.929	0.779	0.720
Blue	0%	0.997	0.995	0.931	0.916	0.781	0.687
	5%	0.998	0.998	0.995	0.995	0.897	0.807
	50%	0.931	0.937	0.940	0.941	0.794	0.703
	100%	0.887	0.922	0.933	0.933	0.774	0.663
Actor holds	0%	0.999	0.999	0.983	0.983	0.819	0.783
	5%	0.999	0.999	0.988	0.988	0.906	0.876
	50%	0.864	0.906	0.901	0.884	0.780	0.792
	100%	0.981	0.982	0.976	0.978	0.918	0.912
Handover	0%	0.093	0.327	0.238	0.223	0.240	0.224
	5%	0.198	0.376	0.291	0.326	0.431	0.438
	50%	0.903	0.816	0.788	0.769	0.743	0.742
	100%	0.997	0.976	0.968	0.921	0.920	0.890

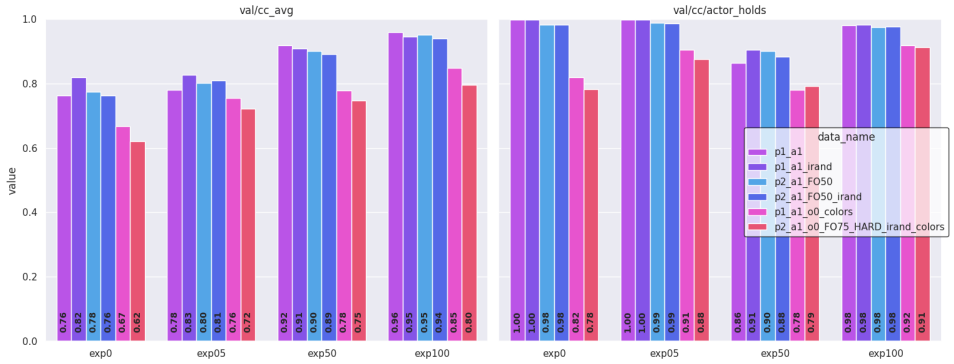


Figure 5: Average CC and per-channel `actor_holds` CC across annotation rates. The average rises with the rate almost entirely because of the `handover` channel; the three core channels are near-perfect already at 0%.

Table 7: `handover` channel localization (NME) across annotation rates. Lower is better; 0.0 is perfect localization. Even at 0% the prediction is centered near the correct cell despite low CC.

Rate	Clean	Irand	4a+FO50	4a+FO50+ir	Colors	Extr.+col
0%	0.056	0.039	0.097	0.105	0.087	0.084
5%	0.040	0.034	0.103	0.091	0.072	0.086
50%	0.008	0.013	0.049	0.043	0.043	0.048
100%	0.000	0.000	0.002	0.014	0.012	0.018

Table 8: Perception quality (IoU@0.5) across annotation rates. Binarizing at 0.5 before computing overlap makes this metric sensitive to prediction confidence rather than spatial correlation.

Channel	Rate	Clean	Irand	4a+FO50	4a+FO50+ir	Colors	Extr.+col
Green	0%	0.983	0.977	0.945	0.900	0.724	0.661
	5%	0.981	0.990	0.951	0.927	0.746	0.688
	50%	0.969	0.984	0.971	0.966	0.773	0.696
	100%	0.966	0.856	0.895	0.898	0.720	0.626
Blue	0%	1.000	1.000	0.869	0.823	0.659	0.559
	5%	1.000	1.000	0.990	0.993	0.794	0.671
	50%	0.834	0.916	0.899	0.898	0.718	0.608
	100%	0.778	0.856	0.856	0.846	0.678	0.573
Actor holds	0%	0.539	0.563	0.686	0.768	0.626	0.553
	5%	0.915	0.859	0.891	0.917	0.743	0.755
	50%	0.772	0.779	0.806	0.809	0.650	0.662
	100%	0.966	0.965	0.955	0.960	0.778	0.673
Handover	0%	0.000	0.300	0.281	0.214	0.188	0.188
	5%	0.000	0.233	0.219	0.167	0.146	0.146
	50%	0.773	0.682	0.609	0.513	0.430	0.407
	100%	0.909	0.886	0.766	0.671	0.590	0.546

A.8 Decision-Level Precision and Recall

Table 9: Handover detection precision (top) and recall (bottom) across all test conditions, complementing Table 2. NeSym rows report the mean over three training seeds, with the sample standard deviation shown below in smaller print; VM baselines are single-run results.

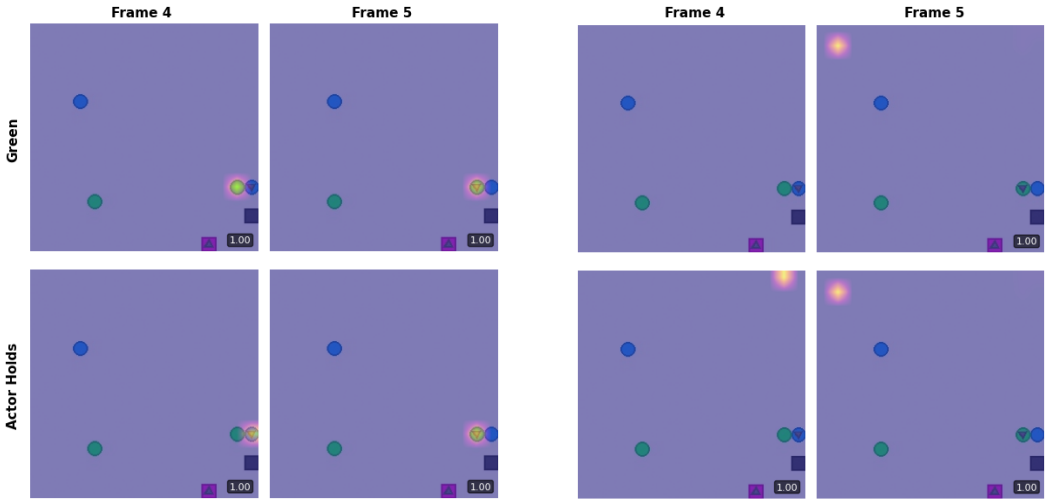
Model	Clean	Irand	4a.	4a+FO	Extr.	Occl.	h30	Colors	Extr.+col
Precision									
NeSym 0%	0.967 ±0.058	0.910 ±0.088	0.921 ±0.137	0.585 ±0.286	0.739 ±0.251	0.978 ±0.038	0.890 ±0.144	0.848 ±0.262	0.625 ±0.115
NeSym 5%	0.967 ±0.058	0.942 ±0.081	0.899 ±0.176	0.645 ±0.345	0.761 ±0.226	0.861 ±0.127	0.883 ±0.140	0.711 ±0.342	0.571 ±0.124
NeSym 50%	1.000 ±0.000	0.955 ±0.018	0.838 ±0.071	0.440 ±0.152	0.620 ±0.133	0.942 ±0.054	0.701 ±0.228	0.706 ±0.098	0.443 ±0.339
NeSym 100%	1.000 ±0.000	0.933 ±0.020	0.887 ±0.169	0.329 ±0.140	0.692 ±0.100	0.868 ±0.229	0.723 ±0.047	0.791 ±0.149	0.496 ±0.200
VM-bin	1.000	1.000	0.923	0.545	0.500	1.000	0.967	0.000	0.000
VM-dir	1.000	0.886	0.211	0.200	0.353	1.000	0.727	0.024	0.200
VM-grid	0.595	0.459	0.488	0.118	0.244	0.490	0.488	0.464	0.378
Recall									
NeSym 0%	1.000 ±0.000	1.000 ±0.000	1.000 ±0.000	1.000 ±0.000	0.910 ±0.046	0.692 ±0.045	0.825 ±0.160	0.792 ±0.361	0.376 ±0.125
NeSym 5%	0.963 ±0.064	0.980 ±0.034	0.958 ±0.072	1.000 ±0.000	0.886 ±0.087	0.634 ±0.154	0.861 ±0.127	0.373 ±0.164	0.201 ±0.187
NeSym 50%	1.000 ±0.000	1.000 ±0.000	0.938 ±0.062	1.000 ±0.000	0.767 ±0.157	0.838 ±0.100	0.909 ±0.085	0.660 ±0.226	0.577 ±0.277
NeSym 100%	1.000 ±0.000	1.000 ±0.000	0.854 ±0.253	0.917 ±0.144	0.824 ±0.167	0.899 ±0.033	0.950 ±0.022	0.552 ±0.048	0.370 ±0.081
VM-bin	0.927	0.350	0.774	0.261	0.057	0.949	0.690	0.000	0.000
VM-dir	1.000	0.775	0.129	0.130	0.171	1.000	0.571	0.027	0.276
VM-grid	0.610	0.425	0.645	0.174	0.286	0.615	0.476	0.351	0.483

A.9 Full Ablation Results

Table 10: Symbolic induction ablation, all test datasets, evaluated via `holds_transfer` except for *VC neural*, which reads the neural handover channel directly. *CP+Scallop* is an intermediate variant using only the color prior alongside the Scallop rules.

Dataset	Full induction	Scallop only	VC transfer	VC neural	CP+Scallop
Clean	1.000	1.000	1.000	1.000	1.000
Objects	1.000	0.933	1.000	0.986	0.886
Occluded	0.829	0.650	0.696	0.886	0.854
h30	0.857	0.701	0.880	0.951	0.528
FO50	0.983	0.967	0.983	0.966	0.906
Irand	0.988	0.938	0.857	0.976	0.988
4-actor	1.000	1.000	0.852	0.906	0.630
4a+FO50	0.902	0.905	0.939	0.585	0.590
4a+HARD	0.909	0.829	0.930	0.667	0.556
Extreme	0.815	0.825	0.765	0.594	0.750
Colors	1.000	0.195	0.000	0.763	0.725
Extr.+col	0.440	0.526	0.000	0.217	0.468

A.10 Additional Result Figures



(a) Full Induction (b) Scallop Only

Figure 6: Cell-level Grad×Activation attribution for the `holds_transfer` rule in the full induction (left) and Scallop-only (right) models. Rows are the `blue` and `actor_holds` channels; columns are consecutive frames around the handover. Full induction localizes on the correct agent cells; Scallop only routes gradient through the same channels but the attribution spreads across the grid.

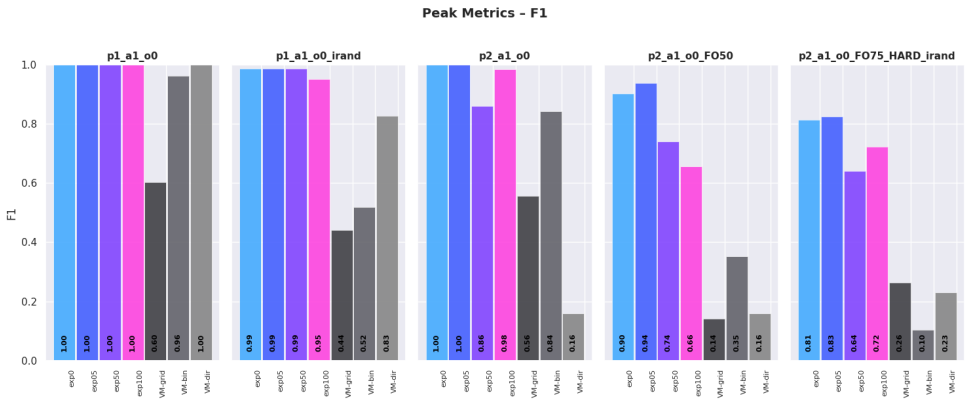


Figure 7: Handover detection F1 across increasing scenario complexity. Neuro-symbolic models at every annotation rate substantially outperform the baselines once scenes leave the training distribution.

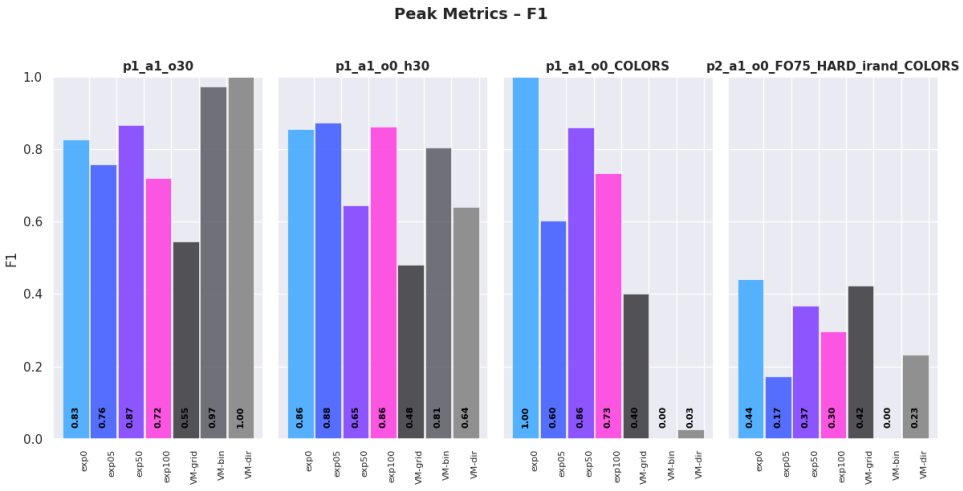
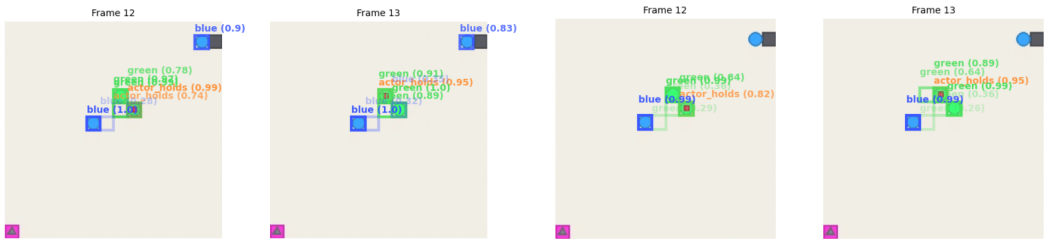


Figure 8: Handover detection F1 under visual and semantic perturbations. The color-transfer sets replace green/blue agents with orange/red, testing generalization beyond the color prior's training palette.



(a) exp100 model cell prediction

(b) exp0 model cell prediction

Figure 9: Cell predictions of the 100% (left) and 0% (right) models on the hard dataset. The 0% model degrades less under distribution shift; here a spurious `blue` activation by the 100% model triggers `holds_transfer` and yields a false positive.

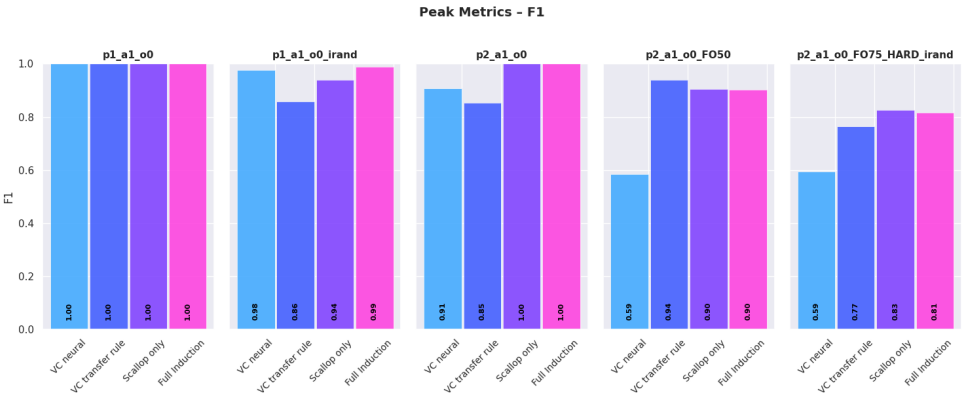


Figure 10: Handover detection F1 across test datasets for the ablation configurations. Both single-function variants track the full model closely on the decision task.