2nd German Robotics Conference
March 11-13, 2026. Cologne, Germany

Entity-Grounded Procedural Knowledge Graphs for Executable Task Understanding from Instructional Videos

Cennet Oguz, Simon Ostermann, Günter Neumann
German Research Center for Artificial Intelligence (DFKI)
{cennet.oguz,simon.ostermann,guenter.neumann}@dfki.de

***Abstract*— Instructional videos contain rich procedural knowledge that could support robotic task execution. However, most existing video understanding approaches produce free-form captions or high-level action labels that lack the explicit, entity-centric semantics required for robotic planning. We present *Entity-Grounded Procedural Knowledge Graphs (EGPKGs)*, a neuro-symbolic representation that decomposes instructional videos into explicit entity-level transformations with grounded preconditions and effects. EGPKGs integrate language-based action schemas, vision-based entity grounding, and symbolic state transitions to produce executable task representations suitable for AI-powered robotic systems.**

I. INTRODUCTION

Instructional videos are a widely available source of procedural knowledge for tasks such as cooking, assembly, and maintenance, and are therefore of growing interest for robotic learning and task understanding [1], [2]. However, despite strong progress in vision and vision-language models, most outputs remain optimized for human interpretability rather than robotic execution, lacking the explicit structure needed for task planning, execution, and monitoring. Robotic execution requires representations that specify which entities participate in each step, how these entities transform over time, and how individual transformations enable subsequent actions within a task [3]. In contrast, free-form descriptions such as "mix the ingredients" or "cook until done" omit crucial information about object identity, state changes, and causal dependencies. Without explicit modeling of entity persistence and symbolic effects, robotic systems cannot reliably verify task progress, reason about preconditions, or recover from failures during execution [12].

Bridging this gap requires representations that connect perceptual observations from video to executable symbolic task models. Recent work has highlighted the importance of combining perceptual learning with symbolic reasoning to enable robust long-horizon robotic behavior [10], [11]. This motivates representations that are both grounded in visual evidence and structured according to the requirements of symbolic planning and execution.

Throughout this work, we use instructional cooking videos as a running example, as they provide rich visual evidence of entity presence, manipulation, and state change across temporally structured actions, making them particularly suitable for studying visually grounded, entity-centric procedural reasoning.

II. ENTITY-GROUNDED PROCEDURAL KNOWLEDGE GRAPHS

We introduce *Entity-Grounded Procedural Knowledge Graphs (EGPKGs)*, a representation that converts instructional videos into sequences of explicit, entity-level state transformations. Each instruction step is modeled as an operator that consumes one or more input entities and produces one or more output entities with updated symbolic states, while preserving entity identity across steps through coreference links. This design is motivated by classical symbolic planning formalisms, where actions are defined in terms of preconditions and effects over world states, but extends them with perceptual grounding in visual observations [5], [3].

EGPKGs integrate three complementary components that correspond to language, vision, and symbolic reasoning. First, large language models predict step-level symbolic structure from natural language instructions, including action schemas and referenced entities, building on recent progress in structured semantic parsing and instruction understanding [6], [7]. Second, vision–language models ground these entities in video frames, providing perceptual evidence for both the availability of input entities and the realization of output states. This grounding step is inspired by prior work on visually grounded language understanding and multimodal alignment [8], [9].

Third, a symbolic layer connects grounded transformations into a causally structured procedural graph that explicitly encodes preconditions, effects, and entity evolution over time. By tracking entity persistence and state changes across steps, EGPKGs address limitations of prior video understanding approaches that focus on action labels or free-form captions without modeling object identity or causal structure [1], [2]. The resulting representation aligns perceptual evidence with symbolic task logic, producing an interpretable and executable procedural model suitable for downstream robotic planning and reasoning.

A. State Space, Operators, and Symbolic Transitions

Let $E = \{e_1, \ldots, e_N\}$ denote the set of entities participating in the task (e.g., `noodles`, `chicken`, `wok`). Each entity e is described by symbolic state variables $\mathsf{Vars}(e) = \{\texttt{state}(e), \texttt{location}(e)\}$, which capture properties relevant for procedural reasoning such as preparation state and spatial configuration. A world state s is a complete assignment of values to all state variables across all entities.

Each instruction step S_i is modeled as a symbolic operator $S_i : \mathcal{S} \rightarrow \mathcal{S}$ that transforms one world state into another. Preconditions $\mathrm{Pre}(S_i)$ specify which symbolic conditions must hold for the step to begin, while effects $\mathrm{Eff}(S_i)$ describe the state changes produced by the action. Causal links between steps arise when the effects of one step satisfy the preconditions of a later step, forming the logical backbone of the PKG.

B. Step Inputs and Outputs

Given an ordered sequence of steps $\texttt{recipe} = (s_0, \ldots, s_n)$ of a recipe video, the LLM predicts a structured representation for each step that identifies participating entities and intended symbolic outcomes. Each step may consume multiple input entities, including ingredients, tools, containers, or intermediate products. Coreference links connect these inputs to their originating entities or states in previous steps, preserving entity identity across the procedure.

A step may produce one or more output entities, including composite entities such as mixtures or coated ingredients. Each output entity is annotated with an intended symbolic state and, when applicable, a location. Together, the input and output structures fully specify the symbolic transformation performed by the step, enabling consistent multi-entity reasoning.

C. VLM-Based Visual Grounding

For each step s_i, the corresponding video segment is processed by a Vision–Language Model [14], [15] to ground symbolic entities in the visual stream. The VLM outputs regions corresponding to input entities and output entities, providing frame-level evidence for both step initiation and completion. Input grounding verifies that required entities are present and in suitable initial states, while output grounding confirms that symbolic effects have been visually realized.

This grounding mechanism enables the detection of delayed or asynchronous effects, such as cooking or resting actions, whose postconditions may be achieved well after the corresponding instruction is issued. By associating symbolic effects with their true visual realization, the PKG avoids reliance on instruction boundaries alone.

D. Graph Construction

The complete PKG is represented as a graph $\mathcal{G} = (V, E)$ comprising step nodes, entity nodes, and state nodes. Temporal edges encode instruction order, consumption and production edges encode preconditions and effects, coreference edges preserve entity identity, and grounding edges link symbolic nodes to visual evidence. Together, these nodes and edges encode the temporal, causal, and referential structure of the procedure.

While the graph construction mechanism is conceptually grounded in prior work on symbolic planning graphs and action representations from classical AI planning, our contribution lies in unifying these strands with visually grounded action understanding into a single entity-grounded procedural graph that explicitly tracks entity identity, state evolution, and visual realization across steps [5], [4], [3]. This design is informed by our prior work [16], [17] on multimodal entity tracking and anaphora resolution in instructional videos, which demonstrated the importance of persistent entity representations aligned with visual evidence across temporally extended procedures [1], [2]. Here, we generalize these insights into a reusable procedural graph formalism, moving beyond a purely conceptual proposal toward a practically instantiated representation that supports execution, verification, and downstream reasoning.

III. RELEVANCE FOR ROBOTICS

Unlike conventional video captions, EGPKGs are directly usable as intermediate representations for robotic systems because they explicitly encode inputs, outputs, and state transitions. This structure aligns with robotic task planning and execution frameworks, where actions must specify preconditions and effects over world states to support planning, monitoring, and recovery [3], [4]. Explicit state modeling enables task planning, execution monitoring, and failure diagnosis, which are difficult to support with free-form text.

By grounding symbolic effects in visual evidence, EGPKGs allow robots to verify whether intended transformations have occurred rather than assuming completion from instruction order alone. This capability is essential for robust real-world execution, where actions may fail or yield partial effects [**?**], [**?**], and supports closing the perception–action loop. The entity-centric structure of EGPKGs enables reasoning about object persistence, delayed effects, and multi-entity actions common in manipulation tasks. By preserving entity identity across transformations, EGPKGs support consistent symbolic reasoning over evolving objects, a key requirement for long-horizon tasks [10]. Overall, EGPKGs bridge perception-driven video understanding and executable robotic task representations, and integrate naturally with symbolic planners and hybrid neuro-symbolic control architectures [11].

IV. CURRENT STATUS AND FUTURE DIRECTIONS

We are developing a multimodal pipeline that combines language models for operator prediction with vision–language models for entity grounding and effect verification. Ongoing work focuses on improving robustness to ambiguous references and delayed effects. Future directions include integrating EGPKGs with robotic task planners, extending the representation beyond cooking scenarios, and evaluating execution fidelity in simulated and real robotic environments.

V. CONCLUSIONS

Entity-Grounded Procedural Knowledge Graphs provide a structured, interpretable, and executable representation of procedural knowledge extracted from instructional videos. By aligning language, vision, and symbolic reasoning at the level required for robotic execution, this work contributes toward making large-scale video knowledge actionable for AI-powered robotic systems.

REFERENCES

[1] L. Zhou, C. Xu, and J. J. Corso, "Towards Automatic Learning of Procedures from Web Instructional Videos," in *Proc. AAAI Conf. Artificial Intelligence (AAAI)*, 2018. :contentReference[oaicite:3]index=3

[2] A. Miech, J.-B. Alayrac, L. Smaira, I. Laptev, J. Sivic, and A. Zisserman, "End-to-End Learning of Visual Representations from Uncurated Instructional Videos," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020. :contentReference[oaicite:4]index=4

[3] M. Ghallab, D. Nau, and P. Traverso, *Automated Planning: Theory and Practice*. San Francisco, CA, USA: Morgan Kaufmann, 2004. :contentReference[oaicite:5]index=5

[4] M. Fox and D. Long, "PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains," *Journal of Artificial Intelligence Research*, vol. 20, pp. 61–124, 2003. :contentReference[oaicite:6]index=6

[5] R. E. Fikes and N. J. Nilsson, "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence*, vol. 2, no. 3–4, pp. 189–208, 1971.

[6] T. B. Brown *et al.*, "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[7] J. Wei *et al.*, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[8] A. Radford *et al.*, "Learning Transferable Visual Models From Natural Language Supervision," in *Proc. Int. Conf. Machine Learning (ICML)*, 2021, pp. 8748–8763. :contentReference[oaicite:7]index=7

[9] J.-B. Alayrac *et al.*, "Flamingo: A Visual Language Model for Few-Shot Learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[10] M. Beetz, D. Beßler, A. Haidu, M. Pomarlan, A. K. Bozcuoğlu, and G. Bartels, "KnowRob 2.0: A 2nd Generation Knowledge Processing Framework for Cognition-Enabled Robotic Agents," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2018, pp. 512–519. :contentReference[oaicite:8]index=8

[11] C. R. Garrett, T. Lozano-Pérez, and L. P. Kaelbling, "PDDLStream: Integrating Symbolic Planners and Blackbox Samplers," in *Proc. AAAI Conf. Artificial Intelligence (AAAI)*, 2020. :contentReference[oaicite:9]index=9

[12] M. Fox, J. Gough, and D. Long, "Detecting Execution Failures Using Learned Action Models," in *Proc. AAAI Conf. Artificial Intelligence (AAAI)*, 2007. :contentReference[oaicite:10]index=10

[13] C. R. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L. P. Kaelbling, and T. Lozano-Pérez, "Integrated Task and Motion Planning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 4, pp. 265–293, 2021. :contentReference[oaicite:11]index=11

[14] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging Properties in Self-Supervised Vision Transformers," in *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, pp. 9630–9640, 2021.

[15] X. Chen, J.-B. Alayrac, C. Riquelme, *et al.*, "PaliGemma 2: A Family of Versatile Vision-Language Models for Transfer," *arXiv preprint arXiv:2412.03555*, 2024.

[16] C. Oguz, I. Kruijff-Korbayova, E. Vincent, P. Denis, and J. van Genabith, "Chop and Change: Anaphora Resolution in Instructional Cooking Videos," in *Findings of the Assoc. for Comput. Linguistics: AACL-IJCNLP 2022*, pp. 364–374, Nov. 2022.

[17] C. Oguz, P. Denis, E. Vincent, S. Ostermann, and J. van Genabith, "Find-2-Find: Multitask Learning for Anaphora Resolution and Object Localization," in *Proc. 2023 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8099–8110, Dec. 2023.