Neuro-Symbolic AI in Computer Vision: Toward More Interpretable, Efficient, Generalized, and Logical Visual Understanding Systems

Original
Neuro-Symbolic AI in Computer Vision: Toward More Interpretable, Efficient, Generalized, and Logical Visual Understanding Systems / Russo, Alessandro; Morra, Lia; Lamberti, Fabrizio. - In: IT PROFESSIONAL. - ISSN 1520-9202. - ELETTRONICO. - 28:1(2026), pp. 26-34. [10.1109/MITP.2025.3627196]

Availability:
This version is available at: 11583/3004590 since: 2026-02-25T09:39:31Z

Publisher:
IEEE Computer Society

Published
DOI:10.1109/MITP.2025.3627196

Neuro-Symbolic AI in Computer Vision: Toward More Interpretable, Efficient, Generalized, and Logical Visual Understanding Systems

Alessandro Russo, *Politecnico di Torino, Torino, TO, 10129, Italy*
Lia Morra, *Politecnico di Torino, Torino, TO, 10129, Italy*
Fabrizio Lamberti, *Politecnico di Torino, Torino, TO, 10129, Italy*

Abstract—Computer vision has evolved dramatically from traditional handcrafted image processing methods to advanced deep learning models. However, despite achieving notable results, these purely statistical methods often suffer from limitations in interpretability, data efficiency, generalization, and reasoning capabilities. Neuro-Symbolic (NeSy) AI has emerged as a promising paradigm that integrates the powerful pattern recognition of neural networks with the structured, logical reasoning of symbolic systems. This paper provides a comprehensive introduction to NeSy applications in computer vision, covering tasks such as image classification, object detection, scene understanding, and action recognition. We explore key NeSy frameworks, including Logic Tensor Networks (LTNs), highlighting their ability to improve interpretability, robustness, and reasoning. Finally, we discuss the challenges and future directions this promising hybrid approach poses toward explainable and trustworthy computer vision solutions.

Computer vision is a subfield of artificial intelligence (AI) that enables machines to interpret and understand visual data from the world. Over time, it has evolved from simple image processing techniques to sophisticated deep learning models capable of performing complex tasks such as object detection, scene segmentation, and facial recognition. Modern computer vision relies heavily on neural-network-based approaches, to extract meaningful representations from images and videos. While modern deep learning methods have achieved remarkable results, they struggle to generalize beyond training distributions, provide little interpretability, and lack explicit reasoning capabilities [17].

Neuro-Symbolic (NeSy) AI has emerged as a promising paradigm that seeks to address these limitations. By integrating statistical learning with symbolic reasoning, NeSy AI enables architectures that combine the powerful pattern recognition capabilities of neural networks with the logical expressiveness of symbolic systems, allowing for models that are not only accurate but also explainable, data-efficient, and logically consistent. Thanks to their ability to perform tasks that require reasoning, recent advances in NeSy AI models have shown promise in various domains [17;3], including natural language processing, robotics, and, specifically, computer vision.

The aim of this work is to explore the principles of NeSy AI, the frameworks that support it, and its applications in the field of computer vision. Unlike previous surveys that focused on offering conceptual overviews on the evolution of NeSy AI systems [17;3], this work provides a more practical and in-depth analysis of specific applications in machine vision, including object detection, scene understanding, and visual reasoning. It also discusses challenges such as scalability, knowledge acquisition, and optimization, highlighting promising directions for future research. Through concrete examples, the aim is to demonstrate the practical value of NeSy approaches and to provide a useful resource for researchers and practitioners seeking to develop more interpretable, robust, and reasoning-capable vision systems.

Digital Object Identifier 10.1109/XXX.0000.0000000

NEURO-SYMBOLIC AI

The promise of NeSy AI is to combine the strengths of Symbolic AI (interpretability and structured reasoning) and Statistical AI (ability to learn at scale and robustness in the presence of noise and ambiguity), while mitigating their respective weaknesses. It also provides a framework to leverage both data and knowledge, such as ontologies and knowledge graphs, in the learning cycle[17]. This section introduces the key principles that underpin recent approaches to NeSy AI.

Bridging System 1 and System 2 Decision Making

An intuitive way to interpret NeSy AI is through the lens of cognitive science, such as the System 1 and System 2 framework popularized by Daniel Kahneman in his book “Thinking, Fast and Slow”[3]:

- *System 1*— This mode of decision-making is fast, automatic, subconscious, and intuitive. It encapsulates pattern recognition and associative learning, making it a useful analogy to traditional deep learning models which process vast amounts of unstructured data efficiently. In computer vision, traditional neural networks like CNNs and transformers exhibit System 1 thinking by rapidly classifying images and recognizing objects. The advantages of System 1 thinking are its speed and efficiency; however, it lacks transparency and can be prone to biases or errors in ambiguous scenarios.
- *System 2*— Unlike System 1, System 2 thinking is slow, deliberate, and logical. It involves structured reasoning and symbolic association. Symbolic AI and logical reasoning frameworks, which offer explainable decision making in AI applications, present a possible implementation of System 2-like reasoning. Although it is more robust and interpretable, System 2 thinking is computationally expensive.

By integrating System 1 and System 2 thinking, NeSy AI creates hybrid models that leverage both intuitive pattern recognition and structured logical reasoning. For example, in autonomous driving, System 1 thinking can detect obstacles quickly, while System 2 thinking evaluates the best course of action based on rules and domain knowledge.

Logic Definition in Neuro-Symbolic AI

Logic plays a fundamental role in NeSy AI, where the most widely used logical paradigms are first-order logic (FOL) and real Logic. FOL, also known as predicate logic, provides a rigorous foundation for knowledge representation and reasoning. Fuzzy or real logic expands traditional FOLs by replacing binary truth values (true or false) with truth values on a continuum between 0 and 1. This approach makes fuzzy logic particularly useful for real-world applications where decisions involve imprecise or ambiguous information. In NeSy AI, fuzzy logic is often used to model symbolic variables and uncertainty in decision-making processes. For instance, in image recognition tasks, fuzzy logic enables AI systems to classify objects with varying degrees of confidence, improving robustness in noisy environments.

Several frameworks have been proposed to integrate neural learning with symbolic reasoning. Examples of prominent frameworks, each with their strengths and weaknesses (summarized in Table 1), are:

- *Logic Tensor Networks (LTNs)*— LTNs[1] embed FOL statements directly into neural networks by representing entities as tensors and logical predicates as differentiable functions. Based on real (fuzzy) logic, they assign continuous truth values in the [0,1] interval, thus representing uncertainty without the need for probabilistic inference. LTNs reformulate the learning process to maximize the aggregated satisfiability of a knowledge base consisting of one or more logical statements. This key feature enables the incorporation of symbolic constraints directly into the learning process, resulting in models that strive for logical consistency while leveraging neural networks for perception and pattern recognition. However, the resulting loss function may have a large memory footprint or be slow to converge, especially in large or complex domains. Current implementations are available in Pytorch[1] and Tensorflow[2].
- *Neuro-Symbolic Probabilistic Soft Logic (NeuPSL)*— NeuPSL[14] extends Probabilistic Soft Logic (PSL) by using using neural networks to generate soft truth values in the [0,1] interval, which act as probabilistic input evidence for a PSL layer. These values are integrated into a set of weighted and differentiable logical rules expressed as convex functions, allowing coupling and joint training of the neural and

[1] https://github.com/tommasocarraro/LTNtorch
[2] https://github.com/logictensornetworks/logictensornetworks

symbolic component. NeuPSL belongs to a class of NeSy systems based on Energy-Based Models; specifically, the symbolic and neural components together are interpreted as a hinge-loss Markov random field, a tractable probabilistic graphical model allowing for end-to-end training. Unlike LTNs, NeuPSL maintains a looser coupling between neural and symbolic components: the neural network produces soft predictions that the symbolic layer reasons over in a separate inference step. This design sacrifices some expressiveness – NeuPSL rules must be in a form compatible with PSL convex relaxation – but gains significantly in scalability. Current implementations are available in Tensorflow[3].

› *DeepProbLog*— DeepProbLog[9] extends the Prolog programming language by integrating neural networks into probabilistic logic programming. Neural networks are used to define the probability distributions over certain facts (e.g., image $\rightarrow$ digit), which are injected into symbolic logic programs. Symbolic reasoning is then performed by a compiler that converts the logic program to a sentential decision diagram (SDD), which provides probabilities for possible explanations. Unlike frameworks such as LTNs or NeuPSL that use continuous-valued logic, DeepProbLog preserves a discrete logical structure. It also supports end-to-end differentiable learning by enabling the gradients to flow through the logic circuit back into the neural networks, allowing for joint optimization of both neural and symbolic parameters. This feature allows DeepProbLog to offer a tighter integration between learning and discrete logic, albeit at higher computational cost due to the symbolic inference. Current implementations are available in Pytorch and Prolog[4].

The Neural-Symbolic Learning Cycle

The NeSy learning cycle[12;13] is a structured, iterative workflow that conceptualizes how knowledge flows bidirectionally between the neural sub-symbolic component and the symbolic component of a NeSy system. It is composed of three interlinked phases: knowledge extraction, user interaction, and knowledge injection. Together, they support continuous learning, interpretability, and domain alignment. An example of application of the NeSy cycle to the task of scene understanding is shown in Figure 1.

Knowledge extraction begins with analyzing the internal representations learned by neural models. For example, neuron activations in a CNN can be systematically mapped to symbolic concepts using ontology-guided tools such as Concept Induction[13;12]. In this phase, interpretable patterns or rules are identified to approximate what the network has learned, such as which neurons respond to semantic categories like “vehicle” or “furniture”. In more structured tasks, scene graphs or attribute maps may be extracted from raw output to provide a symbolic layer that describes objects instances, part-whole relationships, or interactions[16;18;2;11].

User interaction plays a crucial role in validating and refining the extracted symbolic knowledge. Domain experts can inspect proposed rules and concepts, correct misinterpretations, or add contextual priors that are missing from the data-driven model. This step ensures that the symbolic layer remains aligned with human reasoning and domain-specific constraints. Moreover, it supports hybrid human-AI workflows, where symbolic abstractions can be audited and modified without retraining the entire network[7;2;13].

Knowledge injection refers to the process of incorporating symbolic knowledge back into the neural model. One common strategy is to encode symbolic rules as regularizers or auxiliary loss functions during training; this is often done using LTNs[16;18;11**?**;15], which support differentiable reasoning over logical constraints. Another strategy involves modifying the model architecture, such as adding logic-aware modules or graph-based reasoning layers like NeuPSL. In some cases, symbolic knowledge is injected during inference by constraining output spaces or refining predictions with a symbolic post-processing layer[7;19].

This cycle has been successfully applied in several domains. In the context of scene understanding[16;18] — for example, the task illustrated in Figure 1 — knowledge extraction corresponds to the classification of entities and relationships within the neural space. Knowledge injection, on the other hand, is achieved through LTNs, which enforce commonsense logical rules over the predicted entity–relationship triplets. Finally, user interacts with the system by querying and modifying the knowledge base; for instance, an end user may introduce additional constraints on the predicted triplets to reflect changing operational requirements, whereas a machine learning practitioner may monitor the performance of the proposed rules and their impact on training, adapting/modifying them

[3] https://github.com/linqs/neupsl-ijcai23
[4] https://github.com/ML-KULeuven/deepproblog

Framework	Strengths	Weaknesses
Logic Tensor Networks (LTNs)[1]	- Provides strong logical expressiveness through the definition of full FOL statements - Fast inference - End-to-end differentiable - Handles uncertainty via fuzzy training objective - Low barrier to adoption	- Poor scalability to large datasets or complex domains - Does not supports exact probabilistic reasoning - Training can be unstable and memory-intensive
Neuro-Symbolic Probabilistic Soft Logic (NeuPSL)[14]	- Highly scalable due to convex inference. - Effective for relational probabilistic reasoning. - Supports exact probabilistic reasoning	- Restricted expressiveness (rules must be in a form compatible with PSL convex relaxation and cannot be quantified over) - Limited adoption in the literature
DeepProbLog[9]	- Strong expressiveness: supports discrete logic and probabilistic reasoning - End-to-end differentiable - Supports exact probabilistic reasoning - Supports logical program induction from examples	- Inference is computationally expensive - Limited scalability to large probabilistic programs - More sensitive to design of logical structure and neural predicates

TABLE 1. Comparison of strengths and weaknesses of three neuro-symbolic frameworks: LTNs, NeuPSL, and DeepProbLog.

as needed.

One of the key strengths of the neural-symbolic cycle is its modularity. Indeed, many current NeSy AI techniques focus primarily on one phase of the NeSy cycle, typically knowledge extraction or induction. Each phase can be updated or improved independently: better symbolic abstraction techniques can enhance extraction, user-guided tools can streamline interaction, and advances in differentiable logic can improve injection fidelity. As such, the cycle supports an incremental and transparent improvement in AI systems, making it an essential design pattern for reliable and adaptive NeSy architectures.

NEURO-SYMBOLIC AI IN COMPUTER VISION

Recent work in diverse computer vision application fields, from scene understanding to medical imaging, has demonstrated how integrating neural and symbolic components can produce more robust, interpretable, and semantically grounded AI systems.

Classification and Object Detection

Several studies have applied NeSy reasoning to core computer vision tasks such as classification and object detection.

For object detection, Faster-LTN[11] enhances the standard Faster R-CNN pipeline with logical constraints embedded via LTNs. These constraints capture part-whole relationships and class co-occurrence (e.g., “a car has wheels"), guiding the classifier toward more semantically plausible outputs.

FLPN[10] (Fuzzy Logic Prototypical Network) applies a similar logic-augmented architecture to the problem of Zero-Shot Learning (ZSL) classification. It uses class hierarchies and attribute rules to learn mappings between image features and semantic descriptions, enabling recognition of unseen classes. This approach outperforms traditional ZSL methods and demonstrates the ability of symbolic priors to support robust generalization in low-data regimes.

The ability to learn directly from logical constraints is also a powerful tool in semi-supervised and unsupervised settings. Benchmarks such as MNIST-add, which involves learning digit addition from pairs of MNIST images, or clustering tasks provide compelling examples of the potential advantages of NeSy frameworks in this area[1].

Structured Scene Understanding and Spatial Reasoning

Given that most NeSy techniques are based on logic languages and structured knowledge (e.g., knowledge graphs), high-level semantic image interpretation tasks such as Scene Graph Generation (SGG) are one of the most common target for NeSy systems. In SGG, the goal is to extract a structured representation from an image or video that includes objects and their relationships. However, traditional models often struggle with the long-tailed distribution of object categories and predicates, resulting in biased predictions and poor generalization to rare or unseen instances.

NeSy methods like ESRA[16] (nEuro-Symbolic Re-

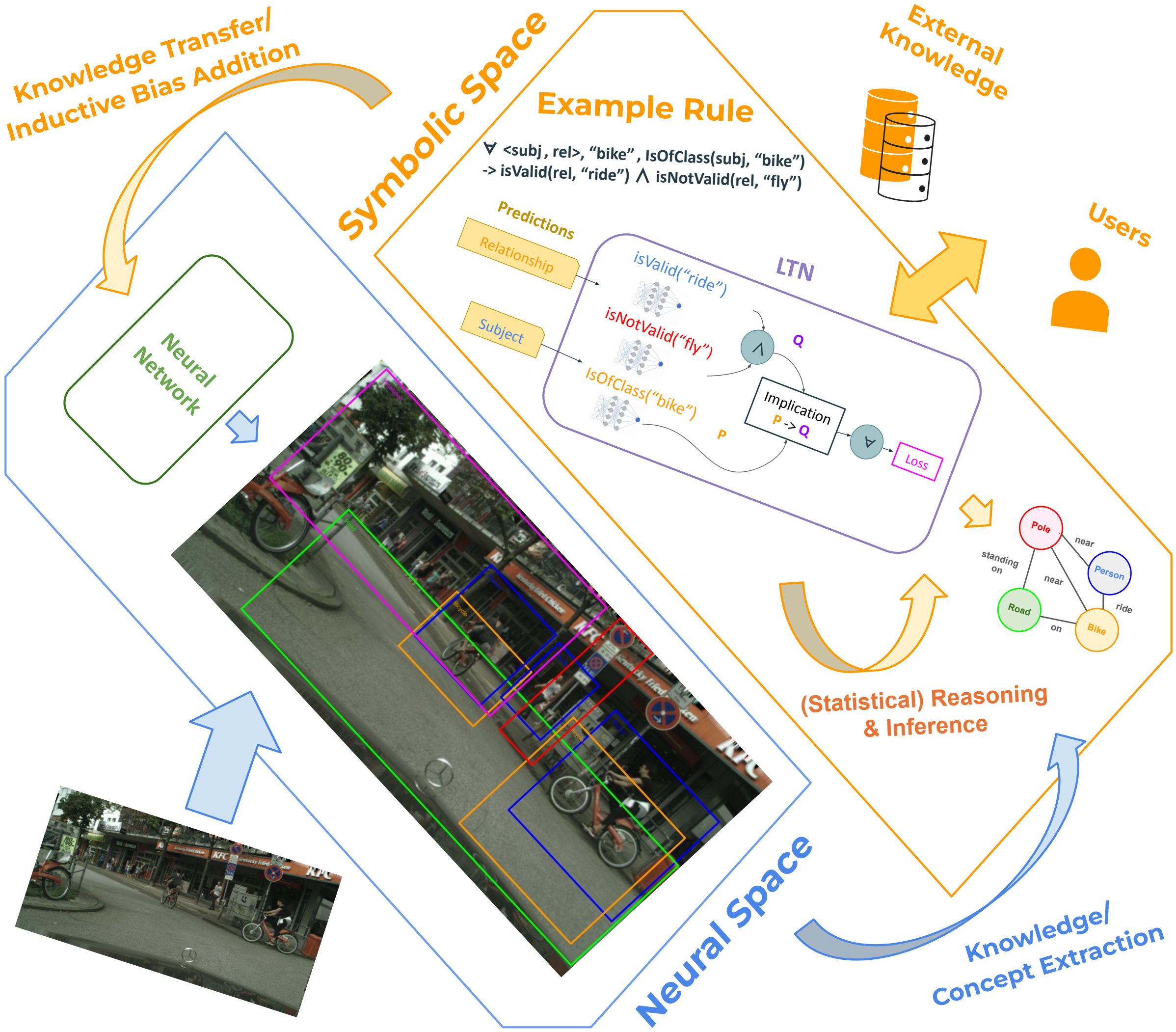

FIGURE 1. Graphical representation of the NeSy cycle applied to the Scene Graph Generation setting, showing how knowledge is transferred between the symbolic and neural spaces in a NeSy system. a) *Knowledge extraction: The scene is first processed by the neural model, generating a set of region proposals along with corresponding predictions (bottom left). b) Knowledge injection: the extracted information is passed to the symbolic layer, where external domain knowledge and user-defined rules are integrated to reason about and validate these predictions, producing the final scene graph (top right). c) User interaction: the output from the symbolic layer is used to analyze the impact of the proposed rules or change constraints to reflect oprational requirements.*

lation trAnsformer) address the challenge of relational reasoning in autonomous driving, where relationships like "car driving on road" or "person riding bike" are crucial for safety. ESRA builds on RelTR, a transformer-based SGG model, and introduces a logic-based loss using the LTN framework. Domain knowledge is encoded as logical axioms, such as constraints on which objects can plausibly serve as subjects or objects in a given relation. This symbolic supervision improves performance, particularly for rare relationships, and ensures that outputs are semantically coherent. Architectures like FLPN[10] and ESRA[16] build on state-of-the-art backbones, such as ViT or DETR and, at inference time, operate like standard deep neural networks: however, their features and predictions benefit from the additional supervision provided by logical constraints during learning.

Another work[18] targets the problem of 3D scene graph generation in both indoor and outdoor environments. The proposed approach integrates Large Language Models (LLMs) with LTNs for spatial relation reasoning. LLMs automatically generate spatial ontologies that are then used to inject symbolic constraints into the learning pipeline. These constraints help disambiguate complex hierarchies and improve generalization to novel environments, enabling the system to learn structured scene representations even from limited or weakly labeled data.

Visual Question Answering (VQA) is another domain where NeSy approaches show strong advantages. The CBNS-VQA[2] (Confidence-Based Neural-Symbolic Visual Question Answering) architecture incorporates uncertainty quantification into structured visual reasoning by combining confidence-aware predictions with a modular NeSy architecture for structured object representation. The framework leverages a neural object perception module with variational dropout to estimate uncertainty over visual attributes and a sequence-to-sequence question parser to generate multiple symbolic programs, each associated with a confidence score. These symbolic programs are then executed over structured scene representations in a symbolic reasoning module, with uncertainty propagated through each reasoning step. In particular, the framework supports human-in-the-loop correction by flagging low-confidence predictions at both the perception and reasoning stages.

Temporal Understanding and Activity Recognition

In the context of Human Activity Recognition (HAR) – where sequences of frames are interpreted to identify actions – a NeSy framework[20] has been proposed to perform context-prompted activity recognition, where the system is guided not only by video frames but also symbolic prompts that list the key objects involved. A graph-based architecture models relationships among body parts and objects, processed via Graph Attention Networks and aggregated temporally via LSTMs. The symbolic object prompts provide improvements to both performance and interpretability, and model errors are semantically closer to the correct class, making them more human-aligned.

In a more challenging setting, ALGO[8] addresses the recognition of egocentric action in an open world setting, where neither the action labels nor the full vocabulary of objects are known at training time. ALGO grounds visual input via an external knowledge base and reasons over object-action graphs to infer plausible activities. This approach enables the system to generalize to novel actions in a zero-shot manner, outperforming closed-world baselines.

Lastly, ViPro[19] targets the task of video prediction under procedural constraints. Standard video predictors often fail in complex, multi-object dynamics due to limited training data or lack of inductive structure. ViPro addresses this challenge by embedding procedural knowledge into the predictor.

Trustworthy, Explainability and Human-Aligned AI

Explainability and human-alignment is a central motivation for NeSy AI. By representing knowledge as logical rules and symbolic abstractions, NeSy systems provide transparency that is often missing in end-to-end neural networks. In high-stakes domains like medical imaging and robotics, NeSy AI is particularly valuable for its interpretability and integration of expert knowledge. Several works exemplify this potential.

SimpleMind[4] is an open-source cognitive AI framework that combines CNNs with a symbolic knowledge base for verifying medical procedures – specifically, the placement of endotracheal tubes in chest X-rays. By grounding the model with medical ontologies and spatial rules, the system achieves high reliability and supports physician trust in its recommendations. Another work[13] proposes a complete NeSy cycle for medical image analysis. The devised framework extracts symbolic rules from a CNN trained to detect pleural effusion in chest X-rays, using the ERIC method to convert convolutional kernel activations into human-readable decision trees (knowledge extractor). A clinician refines these rules to ensure clinical relevance, discarding uninterpretable features or rules inconsistent with medical knowledge. The resulting symbolic rules are then distilled into a smaller, more efficient student CNN (knowledge induction), maintaining high diagnostic accuracy with a notable reduction in parameters. This approach enables end-to-end auditable and interpretable AI pipelines, supporting the design of systems that not only yield correct predictions, but are also “right for the right reasons”.

The Recover[6] system applies similar principles to robotic failure detection and recovery. It uses an ontology to represent the robot environment, logical rules to detect deviations from expected behavior, and an LLM to generate recovery plans. Symbolic supervision ensures that the LLM produces contextually appropriate responses, preventing unrealistic or dangerous plans.

In the diagnostic space, slice discovery[5] is a novel task that identifies subsets of the input space where neural models perform poorly. Inductive Logic Programming (ILP) is used to extract symbolic rules characterizing failure patterns. These rules are used to generate new training data or guide retraining, forming a closed learning loop. This approach improves robustness and fairness by revealing and mitigating hidden model biases.

Frameworks like LTNs are often used to combine labeled data and prior knowledge in the training process, with the latter acting as a soft regularization term. Since the training objective is to maximize overall satisfiability (i.e., to find the best model that satisfies the data under the assumptions provided by the prior knowledge logical statements), logical statements at the end of the training may be partially satisfied or even rejected if in contrast with factual evidence. In many practical applications, this is essential to admit the proverbial exception to the rule, and make rule-based systems robust in real-world applications. However, this behavior may not be ideal in applications where model predictions must comply with predefined standards at all times. For this reason, another line of research[7] focuses on integrating hard logical constraints directly into the training and inference process of deep networks. Unlike soft regularization terms, these constraints are always enforced. Experiments demonstrate that such constraint-aware models generalize better in low-data regimes and offer stronger guarantees in safety-critical applications.

Challenges

Despite the compelling benefits, NeSy methods face several critical challenges that must be addressed for broader adoption in computer vision.

Scalability and Efficiency

Symbolic reasoning methods typically require computationally intensive procedures. Scaling NeSy models to handle high-dimensional visual data or large-scale applications presents a significant barrier, as symbolic reasoning can exponentially increase computational overhead.

Data and Knowledge Acquisition

NeSy AI is highly dependent on accurate symbolic knowledge bases and explicit domain rules. Collecting, curating, and formalizing such structured knowledge can be labor intensive and challenging, particularly for new or complex visual domains where expert knowledge is scarce or expensive to encode. Methods that can induce concepts and/or logical rules with limited human supervision, either directly from data or eliciting them from existing sources such as LLMs, have the potential to increase the applicability of NeSy methods, as envisioned in the NeSy learning cycle.

Optimization and Training

Training NeSy architectures involves balancing symbolic logical constraints with neural network optimization. Achieving stable convergence and optimal performance can be difficult due to conflicting gradients, non-differentiable symbolic components, and challenges related to constraint satisfaction during training[1]. As an example, logical operators and connectives can be grounded by different functions, with different mathematical properties, which may be more or less amenable to stochastic gradient descent[1]; likewise, different formulations of the FOL statements may be equivalent from a logical standpoint but lead to different numerical properties. NeSy losses may also inadvertently introduce reasoning shortcuts, especially in semi-supervised settings.

Implementation

NeSy toolkits, while based on established deep learning framework, are less mature and standardized, which hinders their practical adoption. Many libraries are available to support NeSy applications, especially those in which the symbolic component is fully grounded by neural networks. The software libraries that implement the frameworks (such as LTNTorch[1] or NeuPSL[14]) do not have the level of maturity or support from industry that have reached, e.g., by transformer libraries.

Moving Forward

Addressing these challenges is crucial for realizing the full potential of NeSy approaches and advancing the development of explainable, robust, and intelligent computer vision systems. Future research should focus on simplifying integration techniques by providing standardized benchmarks and toolkits, developing scalable reasoning mechanisms (e.g., by leveraging hybrid training methods), improve knowledge acquisition methods by leveraging automated automated tools (such as LLMs), establishing robust training paradigms that accommodate both symbolic and neural components effectively, and opening toward new fields like generative models.

CONCLUSION

This paper reviewed the use of NeSy AI in computer vision, highlighting its ability to overcome key limitations of traditional deep learning.

Across tasks such as classification, object detection, and scene understanding, these frameworks integrate domain knowledge and enforce logical consistency. Techniques such as LTNs and NeuPSL demonstrate how symbolic constraints can guide learning, while the NeSy learning cycle offers a blueprint for extending these ideas to new tasks.

Despite strong progress, challenges remain in scalability, knowledge acquisition, optimization, and implementation.

Still, momentum is building. NeSy AI provides a promising direction for computer vision systems that not only perceive, but also reason, making decisions rooted in knowledge and logic.

ACKNOWLEDGMENTS

This work was carried out within the FAIR – Future Artificial Intelligence Research and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COM- PONENTE 2, INVESTIMENTO 1.3 – D.D. 1555 11/10/2022, PE00000013). The paper reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

REFERENCES

1. Badreddine, S., Garcez, A.d., Serafini, L., Spranger, M.: Logic Tensor Networks. Artificial Intelligence **303**, 103649 (2022)
2. Bao, Y., Xing, T., Chen, X.: Confidence-based Interactable Neural-symbolic Visual Question Answering. Neurocomputing **564**, 126991 (2024)
3. Bhuyan, B.P., Ramdane-Cherif, A., Tomar, R., Singh, T.: Neuro-symbolic Artificial Intelligence: a Survey. Neural Computing and Applications **36**(21), 12809–12844 (2024)
4. Brown, M.S., Wahi-Anwar, M.W., Choi, Y., Daly, M., Shrestha, L., Wong, K.P., Goldin, J.G., Enzmann, D.R.: Implementing Trustworthy AI in Real-world Medical Imaging using the SimpleMind Software Environment. In: International Conference on Neural-Symbolic Learning and Reasoning. pp. 195–203 (2023)
5. Collevati, M., Eiter, T., Higuera, N.: Leveraging Neurosymbolic AI for Slice Discovery. In: International Conference on Neural-Symbolic Learning and Reasoning. pp. 403–418. Springer (2024)
6. Cornelio, C., Diab, M.: Recover: A Neuro-symbolic Framework for Failure Detection and Recovery. In: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 12435–12442. IEEE (2024)
7. Giunchiglia, E.: Deep Learning with Hard Logical Constraints. Ph.D. thesis, University of Oxford (2022)
8. Kundu, S., Trehan, S., Aakur, S.N.: ALGO: Object-Grounded Visual Commonsense Reasoning for Open-World Egocentric Action Recognition. arXiv e-prints pp. arXiv–2406 (2024)
9. Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., De Raedt, L.: DeepProbLog: Neural Probabilistic Logic Programming. Advances in neural information processing systems **31** (2018)
10. Manigrasso, F., Lamberti, F., Morra, L.: Boosting zero-shot learning through neuro-symbolic integration. Pattern Recognition **170**, 111869 (2026)
11. Manigrasso, F., Miro, F.D., Morra, L., Lamberti, F.: Faster-LTN: a Neuro-symbolic, End-to-End Object Detection Architecture. In: Artificial Neural Networks and Machine Learning–ICANN 2021: 30th International Conference on Artificial Neural Networks. pp. 40–52. Springer (2021)
12. Mileo, A.: Towards a Neuro-symbolic Cycle for Human-centered Explainability. Neurosymbolic Artificial Intelligence **1**, NAI–240740 (2025)
13. Ngan, K.H., Mansouri-Benssassi, E., Phelan, J., Townsend, J., Garcez, A.d.: From Explanation to Intervention: Interactive Knowledge Extraction from Convolutional Neural Networks Used in Radiology. Plos one **19**(4), e0293967 (2024)
14. Pryor, C., Dickens, C., Augustine, E., Albalak, A., Wang, W., Getoor, L.: NeuPSL: Neural Probabilistic Soft Logic. In: International Joint Conference on Artificial Intelligence (IJCAI) (2023)
15. Russo, A., Manigrasso, F., Lamberti, F., Morra, L.: L-TReiD: Logic Tensor Transformer for Re-identification. In: International Symposium on Visual Computing. pp. 345–357. Springer (2023)
16. Russo, A.S., Morra, L., Lamberti, F., Dimasi, P.E.I.: ESRA: a Neuro-Symbolic Relation Transformer for Autonomous Driving. In: 2024 International Joint Conference on Neural Networks (IJCNN). pp. 1–10. IEEE (2024)
17. Sheth, A., Thirunarayan, K.: The Duality of Data and Knowledge Across the Three Waves of AI. IT professional **23**(3), 35–45 (2021)
18. Strader, J., Hughes, N., Chen, W., Speranzon, A., Carlone, L.: Indoor and Outdoor 3d Scene Graph Generation via Language-enabled Spatial Ontologies. IEEE Robotics and Automation Letters pp. 4886 – 4893 (2024)
19. Takenaka, P., Maucher, J., Huber, M.F.: ViPro:

Enabling and Controlling Video Prediction for Complex Dynamical Scenarios Using Procedural Knowledge. In: International Conference on Neural-Symbolic Learning and Reasoning. pp. 62–83. Springer (2024)

20. Xu, B., Bikakis, A., Onah, D., Vlachidis, A., Dickens, L.: Context Helps: Integrating Context Information with Videos in a Graph-based HAR Framework. In: International Conference on Neural-Symbolic Learning and Reasoning. pp. 3–28. Springer (2024)

Alessandro Russo is currently a PhD student at Politecnico di Torino. His current AI interests are in knowledge-infused neuro-symbolic computing and Scene Graph Generation. Contact him at alessandro.russo@polito.it.

Lia Morra is Assistant Professor at Politecnico di Torino. Her research interests include machine learning, image interpretation, and human-computer interaction. Contact her at lia.morra@polito.it.

Fabrizio Lamberti is Full Professor at Politecnico di Torino. His research interests include computational intelligence, human-computer interaction, computer graphics, and visualization. Contact him at fabrizio.lamberti@polito.it.