Perception through Cognitive Emulation

"A Second Iteration of NaivPhys4RP for Learningless and Safe Recognition and 6D-Pose Estimation of (Transparent) Objects"

Franklin Kenghagho K.[1], Michael Neumann[1], Patrick Mania[1] and Michael Beetz[1]

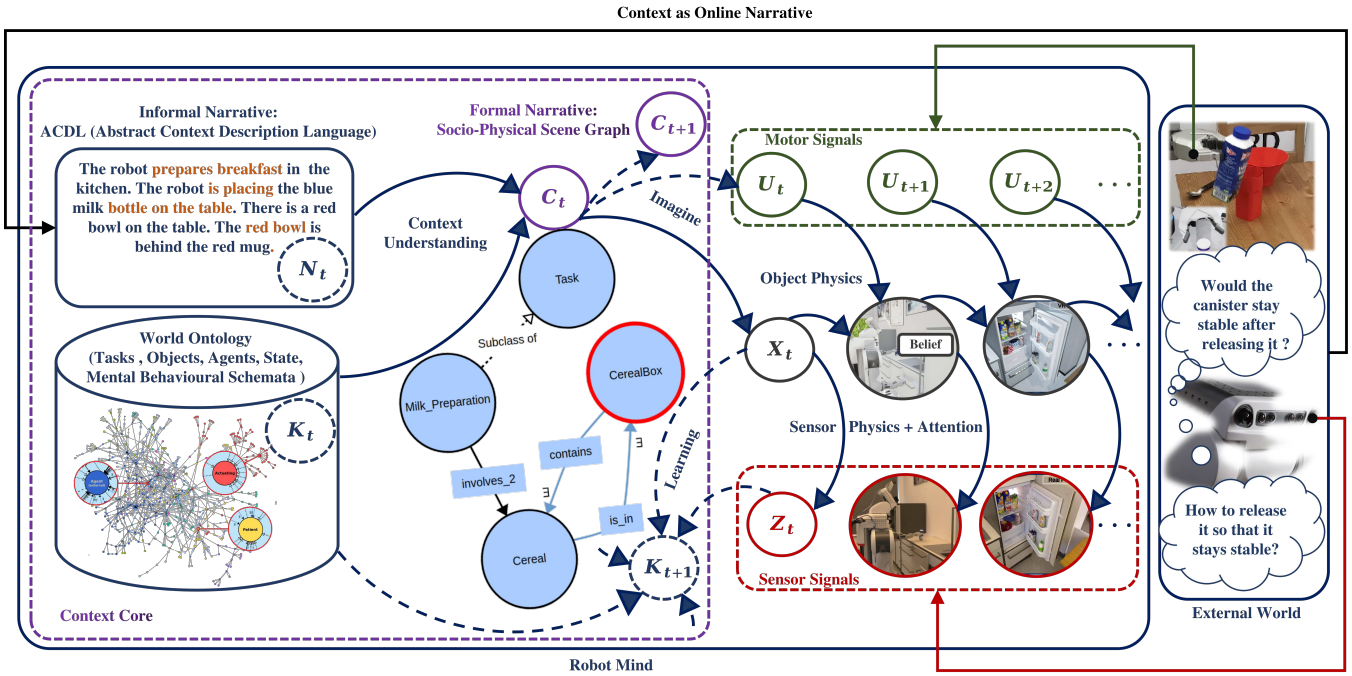

Fig. 1: Sensor data are severely limited in space, time and information for perception in dynamic and mission-critical worlds. Like humans, NaivPhys4RP [6] addresses this issue through cognitive emulation. In this iteration, we present a first complete implementation and demonstrate a learningless and safe recognition and 6D-pose estimation of objects from poor data.

***Abstract*— In our previous work, we designed a human-like white-box and causal generative model of perception NaivPhys4RP, essentially based on cognitive emulation to understand the past, the present and the future of the state of complex worlds from poor observations. In this paper, as recommended in that previous work, we first refine the theoretical model of NaivPhys4RP in terms of integration of variables as well as perceptual inference tasks to solve. Intuitively, the system is closed under the injection, update and dependency of variables. Then, we present a first implementation of NaivPhys4RP that demonstrates the learningless and safe recognition and 6D-Pose estimation of objects from poor sensor data (e.g., occlusion, transparency, poor-depth, in-hand). This does not only make a substantial step forward comparatively to classical perception systems in perceiving objects in these scenarios, but escape the burden of data-intensive learning and operate safely (transparency and causality — we fit sensor data into mentally constructed meaningful worlds). With respect to ChatGPT's ambitions, it can imagine physico-realistic socio-physical scenes from texts, demonstrate understanding of these texts, and all these with no data- and resource-intensive learning.**

*This work was not supported by any organization

[1] with Institute for Artificial Intelligence, Mathematics and Computer Science, University of Bremen, Germany fkenghag@uni-bremen.de

I. INTRODUCTION

Imagine the robot operating in a multi-purpose human-centered environment (i.e, kitchen, lab) such as shown by figure 2. In such contexts, the perception system should sufficiently inform (at least object recognition and 6D-pose estimation) the robot control program about the state of the world so that the latter can act to achieve the desired effects while avoiding the undesired ones. However, this raises at least three challenges. Foremost, **(challenge 1)** notice that the objects in combination with the embodiment of perception produce poor data (e.g., occlusion, no depth) due to interactions (e.g., grasping, clutter) or object materials (e.g., transparent, mirror-like). Unfortunately, this only impairs learning and data interpretation [1]. Secondly, **(challenge 2)** given that objects are permanently out the field of view of the robot's sensors and subject to motion independently of the robot's will, the robot should permanently track the behaviour of these objects in order to anticipate their poses and therefore avoid failures (e.g., fall, spillage). Visual servoing can only react to failures, but not anticipate them. Finally, **(challenge 3)** notice that the tasks such as kitchen

and medical activities are mission-critical in the sense that failures during execution might even cost human life. For this reason, safety is required, and our perception system should ensure that he understands its own outputs (e.g., why is the object a fork rather than a knife?). Despite the

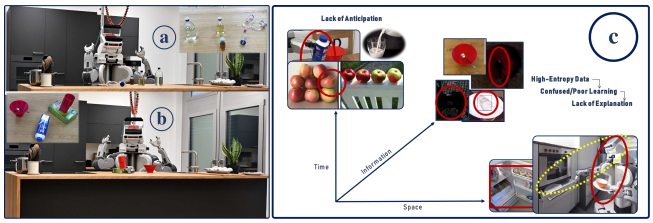

Fig. 2: Challenges of robot perception in dynamic and mission-critical environments (c), (a) medical activity, (b) cooking activity.

prestigious achievements of deep learning, its integration in such applications is impaired due to a lack of transparency and causality. We designed NaivPhys4RP [6] (Naive Physics for Robot Perception) to regard the world state as a situated partially-observable hidden markov process and address such problems mentioned above, by essentially relying on cognitive emulation, in which the target robot permanently emulates how the world state evolves based on commonsense (Causality, Physics, Teleology, Intentions and Utility) [14] in order to understand the past, the present and the future of the world state. Since the emulations are probabilistic, the agent adjusts its emulations based on the few available information from sensor data. As recommended in that previous work, this paper contributes to the improvement of NaivPhys4RP in four significant manners by:

- **refining and generalizing the architecture of NaivPhys4RP** by integrating four key changes (see figure 1): **(a)** a new dynamic cognitive variable N_t for narrative cognition, **(b)** the dynamization of the world ontology variable K_t , **(c)** a transition model of the formal context variable C_{t+1} and **(d)** the addition of an attention mechanism in the sensor model of NaivPhys4RP.
- **delivering** a **first complete implementation of NaivPhys4RP** for anticipating the state and observation of solid worlds, as well as explaining the observations of those worlds.
- **demonstrating** this implementation on **learningless and safe recognition and 6D-Pose estimation of objects from poor sensor data**
- **complementing ChatGPT three steps further**, namely (a) physico-realistic imagination of socio-physical scenes from texts, (b) demonstrating understanding of the texts and (c) with no data- and resource intensive learning.

II. RELATED WORK

There are countless many work papers on recognition and 6D-pose estimation of objects with the most relying on deep learning technologies. However, only few can truly serve embodied agents such as robots in these complex scenes.

Active Perception. Note that most of these approaches consist in moving the robot to better see, which is not only sometimes impossible (moving) but does neither address the challenge 1 as far as data-intensive learning is concerned nor the challenge 2, nor 3 [8, 12, 2, 13, 10].

Visual Servoing. Beyond challenges 1 and 3 which are dismissed, these approaches can w.r.t. challenge 2 only be reactive (even questionable regarding execution speed) but not anticipative: they can detect a failure (e.g., the cup spills) and stop but cannot prevent or avoid it. [5].

Embodied Synthetic Data-based Perception. This approach tries to train systems on synthetic data collected by virtual agents for the sake of sample representativeness. But notice that beside ignoring the challenges 2 and 3, this approach still suffers from transfer learning and intrinsic lack of information in data. [7].

Cognitive Emulation. There are more and more evidences that biological agents (e.g., humans) do not simply rely on sensor data and perform a bottom-up processing of those data to perceive their environments, but rather maintain a mental commonsense apparatus that enables them to emulate the way the world evolves and so understand it. Very important are the key principles identified which drive these mental apparatus namely causality (i.e., cause-effect relations), teleology (e.g., how activities and objects socially relate to each other), utility (i.e., preferences, values of agents), intentions (i.e., goals, actions of agents), and finally physics (i.e., material motion and transformation). Though most of the few available past works in this direction are very shallow in implementing these concepts [8], we theorized with proofs of concept a fully-specified attempt to exhaustively and deeply capture these concepts in a model coined NaivPhys4RP [6].

III. SYSTEM DESCRIPTION

In this section, we present the changes performed on NaivPhys4RP's architecture and present the implementation by describing algorithmically the architecture's component. For further information regarding this implementation, see the GitHub repositories[1].

A. Architecture Refinement

Note that though this subsection relates to key components of the system, these will only be presented in details in next sections.

1) Dynamic Ontology: As presented in NaivPhys4RP's architecture figure 1, the ontology K_t models the fundamental truths about environments, objects, agents, actions and tasks in the intended robot world. Dynamizing this variable makes it flexible (i.e., adding new knowledge), allows it to evolve with the robot experience and enables therefore autonomy (e.g., learning) and flexibility (i.e., cannot add new fundamental knowledge).

2) Context Narrative: Research [9] shows that humans think the course of the ongoing world at a high-level of semantics in terms of narratives. They frame their intentions, goals, observations and tasks in terms of stories. These

[1] https://github.com/NaivPhys4RP

stories allow them to consistently and easily unfold the way the world progresses with very few evidences. This is a special approach for police detectives. For instance, if you are told that people are in a room around a table and eating spaghetti, then you will be able from this basic a narrative to mentally unfold how that room could be like in terms of objects, humans, spatial configurations of these and activities. We introduce a new dynamic variable N_t to model such narrative that allow the robots to roughly frame its thought about the actual context of its environment. That narrative can also come from external agents (i.e., cooperation, collaboration).

3) Smooth Context Transition: The actual robot activity context was only explicitly modelled by the variable C_t, which is in some sense already a formal socio-physical graph of the actual scene highlighting potential objects, their properties, agents, their activities and how those objects relate to those activities. Given that graph C_t, the agent can then imagine a physico-realistic scene X_t as well as actions U_t that correspond to that graph. In order to avoid inconsistent transitions such as moving from the kitchen in time t to toilet in time $t+1$, this paper provides a transition model for this variable and which also allows the incremental specification of the context narrative as the robot performs.

4) Attention-enabled Sensor Model: Once the robot has imagined multiple possible physico-realistic scenes, it filters the most likely ones based on the few available real sensor data Z_t. However, it is not just enough to compute the semantic distance between Z_t and the imagined physico-realistic sensor data Z_t^i since semantically identical images may have drastically different pixel distributions as semantically different images. Therefore, we adjust NaivPhys4RP's sensor model beside the sensor physics by adding an attention mechanism based on Gestalt principles.

5) Inference Tasks: The new system of equations that holds the bayesian inference tasks (markov-blanketed) to solve as anticipatory and explanatory perceptual tasks, described and explained in details in next sections, follows:

$$\begin{cases} \boldsymbol{C}_t^* \sim \boldsymbol{P}(\boldsymbol{C}_t|\boldsymbol{N}_t,\boldsymbol{K}_t,\boldsymbol{C}_{t-1}) & \text{, context understanding} \\ \boldsymbol{X}_t^* \sim \boldsymbol{P}(\boldsymbol{X}_t|\boldsymbol{U}_{0:t-1},\boldsymbol{Z}_{0:t},\boldsymbol{N}_{0:t},\boldsymbol{K}_{0[:t]}) & \text{, actual belief (filtering)} \\ \boldsymbol{X}_{t+1}^* \sim \boldsymbol{P}(\boldsymbol{X}_{t+1}|\boldsymbol{U}_t,\boldsymbol{X}_t,[\boldsymbol{C}_{t+1}]) & \text{, state anticipation} \\ \boldsymbol{X}_{t+1}^*,\boldsymbol{U}_t^* \sim \boldsymbol{P}(\boldsymbol{X}_{t+1},\boldsymbol{U}_t|\boldsymbol{U}_{t+1},\boldsymbol{C}_{t:t+2},\boldsymbol{X}_t,\boldsymbol{X}_{t+2},[\boldsymbol{Z}_{t:t+2}]) & \text{, state explanation} \\ \boldsymbol{Z}_{t+1}^* \sim \boldsymbol{P}(\boldsymbol{Z}_{t+1}|\boldsymbol{X}_{t+1}) & \text{, observation anticipation} \\ \boldsymbol{X}_{t+1}^* \sim \boldsymbol{P}(\boldsymbol{X}_{t+1}|\boldsymbol{U}_t,\boldsymbol{X}_t,\boldsymbol{Z}_{t+1},\boldsymbol{C}_{t+1}) & \text{, observation explanation} \\ \boldsymbol{K}_{t+1}^* \sim \boldsymbol{P}(\boldsymbol{K}_{t+1}|\boldsymbol{U}_{t-1},\boldsymbol{X}_t,\boldsymbol{Z}_t,\boldsymbol{K}_t,\boldsymbol{N}_t,\boldsymbol{C}_t) & \text{, learning} \end{cases} \quad (1)$$

- X, is the world's hidden state (e.g., a digital twin)
- Z, is the object/world observation (e.g., rgbd images)
- U, is the motion control (e.g., joint values, forces)
- K is the world ontology (e.g., world knowledge)
- N, is a narrative describing actual context
- C, is the formal actual context (e.g., knowledge graph)
- $[]$, $\sim$ and * mean optional, sampling, sampled value

B. Scene Ontology Definition (K_t)

Our ontology, written in OWL (Ontology Web Language) extends the SOMA ontology (Socio-Physical Models of Activities) [4] by defining concrete predicates for the kitchen and sterility medical lab activities (See github repo). SOMA, as indicated by the name, captures the physical as well as the social context of objects, agents and agents' actions in everyday activities. Figure 3 illustrates the ontology followed by its key aspects. SOMA is mainly a *taxonomy*, i.e., a

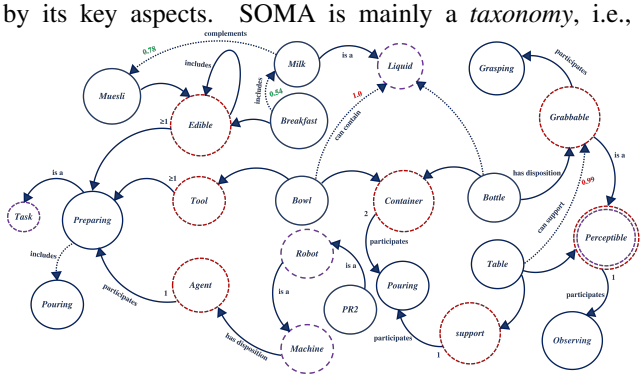

Fig. 3: Illustration of our concrete SOMA-based ontology for kitchen and medical domains. The numbers on the arrows model either a cardinality(B), probability(R) or fuzziness(G).

hierarchy classification (is-a) of generic concepts that we extends with concrete concepts from the kitchen and medical activities. This is important for reasoning about synonyms. The ontology also defines *generic execution plans of actions* as their social contexts by specifying inclusion and precedence relations among actions as well as participants to these actions. Then, it socially describes these *action participants in terms of dispositions*, where an object disposition is a quality which allows the object to take part in an action or event (e.g., knife can cut). In contrast to disposition, *affordance* characterizes how given their dispositions, *objects can interact in a given context* (e.g., cartoonish container cannot hold water).

```
def move_robot_left(self, steps=0.3, step=0.01, topic=0, parallel=True,progressive=True):
    if progressive:
        while (steps > 0):
            steps -= step
            if self.robot_pose.pose.pose.position.y + step > self.robot_base_upper_bounds[1]:
                self.robot_pose.pose.pose.position.y = self.robot_base_upper_bounds[1]
            else:
                if self.robot_pose.pose.pose.position.y + step < self.robot_base_lower_bounds[1]:
                    self.robot_pose.pose.pose.position.y = self.robot_base_lower_bounds[1]
                else:
                    self.robot_pose.pose.pose.position.y += step
```

Fig. 4: Concrete Program for turning the robot base to left.

Behaviour or motion is achieved by grounding primitive actions into primitive parameterized (e.g, by how many degrees should turn head left) symbolic programs operating at the joint value level called *mental behavioural schemata* (see Figure 4). Finally, since OWL does not support probabilistic and fuzzy knowledge representation, we handle fuzziness and uncertainty about concepts implicitly as knowledge in OWL and explicitly extend the main reasoner KnowRob that comes with SOMA [3]. For instance, we may deduce that object A is near to object B from KnowRob, but resolve the degree of proximity in an extra module as explained below.

C. Abstract Context Description Language (ACDL) (N_t)

We define an abstract context description language (ACDL) for flexibly representing the context narrative.

ACDL is a subset of natural language (e.g., English) and can be incremented with the advancement of the reasoning capabilities (e.g., understanding complex-structured sentences). Below is a short overview of ACDL's grammar.

```
context: (statement delimiter)*
statement: subject verb object
object: [determinant] (adjectiv)* noun
subject: [determinant] (adjectiv)* noun
determinant: DET
verb: iverb | dverb
iverb: dverb preposition
noun: NOUN
delimiter: FULLSTOP | COMMA | CONJUNCTION
dverb: DVERB
preposition: PREP
adjectiv: COLOR|SIZE|SHAPE|MATERIAL|TIGHTNESS|mass
mass: NUMBER MASS_UNITS
```

As key features of ACDL, one can note that it enables the expression of narrative in terms of *description of socio-physical contexts of activities*, it has a *recursive grammar* which allows to incrementally make the narrative available to the system and the system to incrementally and recursively process it. Moreover, this recursivity at the statement level of the narrative favours the *smooth transition between contexts* as well as the *implicit expression of negation* (i.e., not such **not** or **no** but semantically overdominating statements causes the forgetting of overdominated ones). Finally, the *language vocabulary and grammar are grounded into the ontology* so that extending the ontology also extends the grammar and the narrative itself can be easily parsed and understood.

D. Context Understanding ($C_t|C_{t-1},N_t,K_t$)

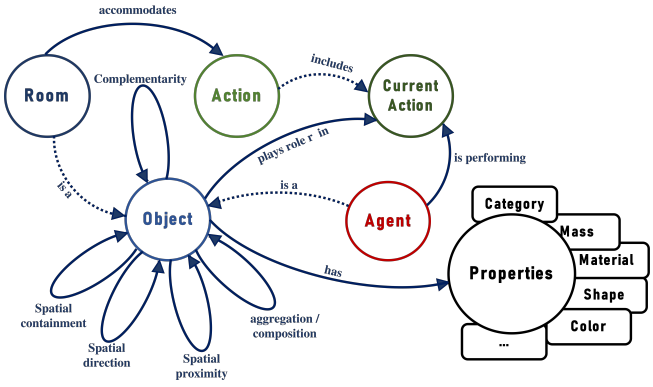

Fig. 5: Structure of socio-physical graphs of robot scenes.

Intuitively, context understanding is about computing the most likely socio-physical ready-to-render (i.e., complete) graphs C_t of the actual scene X_t which are *statistically sufficient* (i.e., no more information is required given this) for X_t given the context narrative N_t, the ontology K_t and the previous graph C_{t-1}. Figure 5 highlights the structure of a socio-physical scene graph. The problem is formalized as a sampling problem (handling uncertainty) and is specified by the first equation of the equation system (1) above namely: $\boldsymbol{C_t^* \sim P(C_t|N_t,K_t,C_{t-1})}$. The sampling algorithm solely based on symbolic reasoning outputs the M most likely graphs $C_t^{(i)}$, a sampling weight/probability as well as an explanation for each of these graphs.

Algorithm 1 Context Understanding: $C_t^* \sim P(C_t|N_t,K_t,C_{t-1})$

Require: $(C_{t-1},w_{t-1},e_{t-1})$, *previous graph, its weight and explanation or empty value*
m, number of graph samples to return
p, number of graph element samples in case of uncertainty
MaxIter, for terminating when ensuring graphs are semantically stable
N_t, *the actual context narrative*
K_t, *the actual world ontology*

Ensure: $S_t = \{(C_t^{(1)},w_t^{(1)},e_t^{(1)}),..,(C_t^{(m)},w_t^{(m)},e_t^{(m)})\}$, *list of m graphs C, each with its explanation e and sampling weight w. e is a text, C is a list of triplets, w is a positive real not greater than one*

$S_t \leftarrow \{(C_{t-1},w_{t-1},e_{t-1})\}$; $nbIter \leftarrow 0$; // *init.,* $C_0 = \emptyset, w_0 = 1.0, e_0 = '', S_0 = \emptyset$
$(R_1,E_1) \leftarrow SyntacticParsing(N_t,K_t)$; // *statement triplets + syntactical structure*
$(R_2,E_2) \leftarrow SymbolGrounding(R_1,K_t)$; // *grounded triplets + grounding table*
if $S_t = \emptyset$ **then**
  $S'_t \leftarrow \{(R_2,1.0,E_1 \oplus E_2)\}$; // *start triplets + max weight=1.0 + explanation*
else
  $S'_t \leftarrow \{(C_{t-1} \cup R_2, w_{t-1}, e_{t-1} \oplus E_1 \oplus E_2)\}$; // *triplets + explanation to old graph*
end if
$stable \leftarrow (S'_t == S_t)$; // *check semantic stability of graphs*
while $\neg stable$ **and** $nbIter < MaxIter$ **do**
  $S_t \leftarrow S'_t$; $nbIter \leftarrow nbIter + 1$; // *update old graph and increment*
  $S'_t \leftarrow InferRoleInEvent(S'_t,K_t)$; *//for each graph g in S', infer generic map of participant roles with multiplicity for each action/event from ontology. The map is added to explanation and weights eventually updated*
  $S'_t \leftarrow InferPotentialRolePlayer(S'_t,K_t)$; *//potential role players for events*
  $S'_t \leftarrow RefineConceptDefinition(S'_t,K_t)$; *//concept refinement: decomposition*
  $S'_t \leftarrow ResolveCorefence(S'_t,K_t)$; *//entities's classes of equivalence*
  $S'_t \leftarrow ResolveEventParticipant(S'_t,K_t)$; *//assignment of participants to events*
  $S'_t \leftarrow ResolveSpatialContainment(S'_t,K_t)$;*//resolve spatial containment(in/on)*
  $S'_t \leftarrow ResolveSpatialDirection(S'_t,K_t)$; *//resolve spatial direction (left/right)*
  $S'_t \leftarrow ResolveSpatialProximity(S'_t,K_t)$; *//resolve spatial proximity (near/far)*
  $S'_t \leftarrow ResolveObjectProperties(S'_t,K_t)$; *//resolve object properties*
  $stable \leftarrow (S'_t == S_t)$; // *check semantic stability of graphs*
end while
$S'_t \leftarrow NormalizeGraphWeight(S'_t)$;*//for each graph g in S', normalizes weight w*
$S_t \leftarrow \emptyset$; // *reset S*
for i=1:M **do**
  $S_t \leftarrow S_t \cup \{WeightedGraphSampling(S'_t)\}\emptyset$; // *sampling M graphs*
end for

Conceptually, Algorithm 1, such as described above, is on the one hand formulated in logical queries which are solved by the reasoning engine KnowRob, which itself builds on top of Prolog in terms of query language and reasoning algorithms. However, since KnowRob like Prolog is unable of probabilistic reasoning, our algorithm performs an explicit probabilistic sampling when KnowRob returns multiple potential solutions for a given query, leading therefore to multiple graphs and an estimation of each graph's weight (M random samples based on weights). Notice that the generation of these graphs constitutes a tree whose leaves are final possible graphs and branches are samples of graph elements with associated sampling probabilities. Then, the weight (also sampling probability) of a graph is the product of all the branches' sampling probabilities in the path between that graph and the root of the tree. Note that this is also similar when resolving fuzziness of concepts. For instance, KnowRob returns that the milk can be in the bottle, bowl or in the cup and near the spoon. Our algorithm will explicitly sample a container and specify the degree of proximity to the spoon. Sampling can be performed in the simplistic case uniformly or in a sophisticated manner while asserting in the ontology the suitability of certain relations (e.g., probability of milk being in bottle higher than being in cup). This also holds when sampling the properties of an object. Additionally, since the algorithm is transparent and symbolic, the explanation results directly from the outputs of each key sub-step. Finally, we presented as many details

as possible in the specification of the inputs/outputs of the algorithm as well as how they are updated in the first line of code.

E. Scene and Action Imagination ($X_t|C_t$, $U_t|C_t$)

Given the computed socio-physical graph of the scene from the context understanding, scene imagination consists, as shown by Figure 6, on the one hand in projecting this graph into a physico-realistic virtual scene X_t through a game-engine-agnostic compiler that converts natural but structured descriptions of objects into simulation engine language (i.e., mesh, material, texture, pose) and in generating the program for U_t corresponding to the specified actions to animate the agent, which in turn will drive the scene X_t in the world transition model.

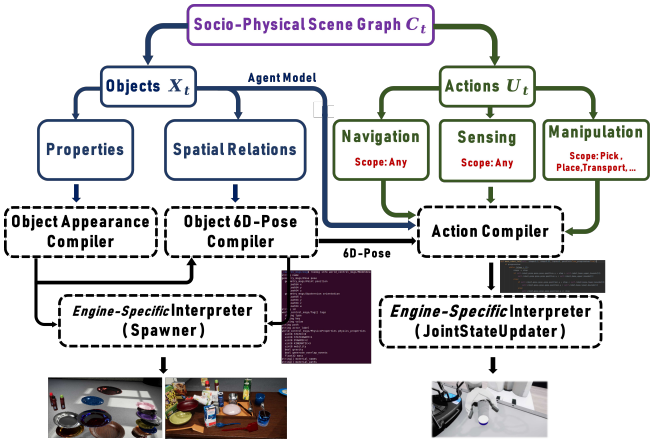

Fig. 6: Circuitry of Scene and Action Imagination.

Then, given a concrete virtual simulation engine such as Unreal Engine, engine-specific interpreters will spawn the objects in the environment and update the joint states of the virtual agents for motion. Formally, the problem to solve is given by the third equation of the system equation (1) namely $\boldsymbol{X}^*_{t+1} \sim \boldsymbol{P}(\boldsymbol{X}_{t+1}|\boldsymbol{U}_t, \boldsymbol{X}_t, [\boldsymbol{C}_{t+1}])$ on the one hand as scene initialization (i.e., $t+1=0$) and on the other hand as a mechanism to initialize incrementally large scenes in parallel with the traditional scene state prediction given by $\boldsymbol{X}^*_{t+1} \sim \boldsymbol{P}(\boldsymbol{X}_{t+1}|\boldsymbol{U}_t, \boldsymbol{X}_t)$ which is presented in the next section (see [6]).

F. Scene Transition Model ($X_{t+1}|U_t, X_t, C_{t+1}$)

Once the scene state and eventually ongoing actions have been imagined in a physico-realistic manner, the ongoing physics in the scene is realized to predict the scene state evolution, which then computes the prediction problem $\boldsymbol{X}^*_{t+1} \sim \boldsymbol{P}(\boldsymbol{X}_{t+1}|\boldsymbol{U}_t, \boldsymbol{X}_t, [\boldsymbol{C}_{t+1}])$. However, notice on the one hand that our predictions are probabilistic due to uncertainty (**e.g., from physics**) which is modelled through a set of simulation particles representing the probabilistic distribution of states (we developed UParaSIM) and on the other hand, the robot can emulate its own ongoing actions U^r_t or imagined actions U^i_t requiring therefore a synchronization mechanism (see Figure 7). Finally, the state prediction with a context variable in contrast to traditional state prediction (i.e., no context variable) enables the incremental initialization of the scene as non initialized areas of the scene can be informed later through an input of the context narrative, then an understanding of this narrative and a subsequent imagination of those areas of the scene which are then merged with the actual scene. This incremental initialization the scene can

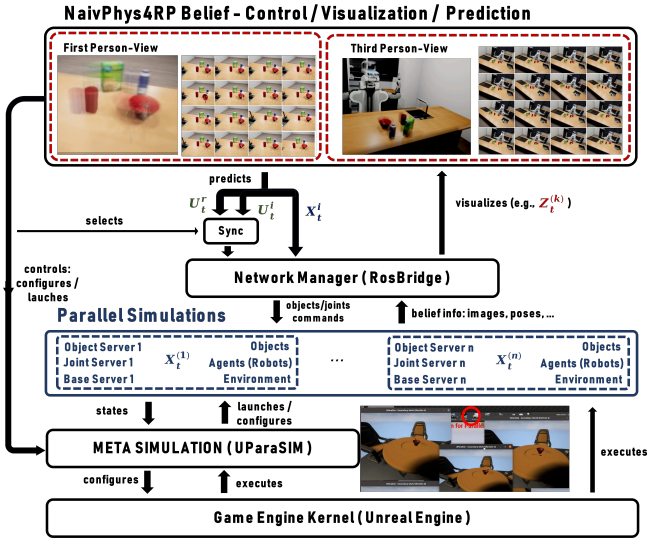

Fig. 7: Probabilistic anticipation in complex scenes and incremental state initialization through imagination.

be regarded as a scene exploration and prevents intractable amount of particles for a complete initialization.

G. Observation Explanation: Recognition and Pose ($X_t|Z_t$)

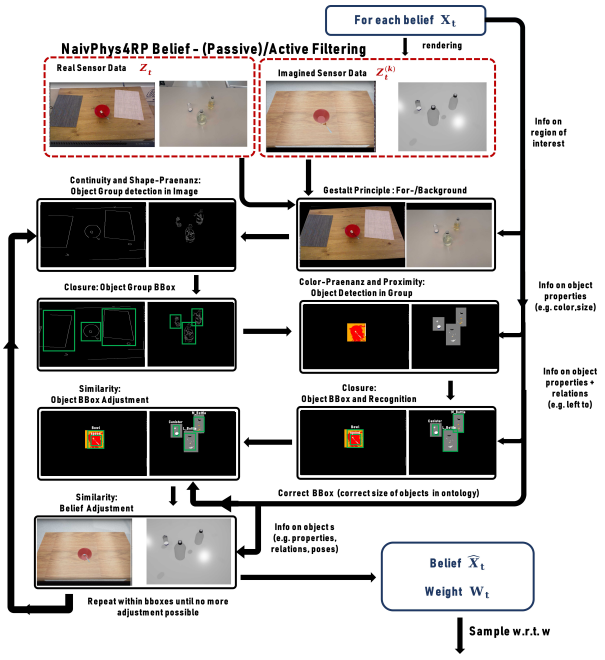

Fig. 8: Recognition and 6D-pose through emulation filtering based on Gestalt principles.

Once NaivPhys4RP is able to generate and realize emulations to predict the evolution of the world state, it should repair or filter or adjust these uncertain emulations $X^{(k)}_t$ given the few available sensor data Z_t. And finding out the world

state $X_t^{(k)}$ that causes at best an observation Z_t is known as observation explanation and is formally given by the fourth equation of the equation system (1) above namely $\boldsymbol{X}^*_{t+1} \sim \boldsymbol{P}(\boldsymbol{X}_{t+1}|\boldsymbol{U}_t, \boldsymbol{X}_t, \boldsymbol{Z}_{t+1}, \boldsymbol{C}_{t+1})$. For each emulation $X_t^{(k)}$, we actively try to fit the sensor data Z_t into $X_t^{(k)}$ based on the Gestalt principles, then compute the error (e.g., no red object found as suggested by $X_t^{(k)}$) and effort (e.g., number of iterations) of the fit as the weight of $X_t^{(k)}$. Finally, we sample the emulations based on their weights. Given the very restricted space allocated for writing the paper, we schematize the algorithm with concrete examples.

IV. Experimentation

As the robot starts performing, it configures the emulator by registering the robot's extero- and interoceptive sensors, the robot's kinematic model, the world ontology, the world's model of physics, the specific language model for ACDL, the number of belief particles and the channels to these data sources (see Figure 9.1). Then, let assume ***the robot prepares the breakfast. The spoon is in the red plastic bowl and the machine looks at the bowl. The robot is in the fridge.***. Figure 9.2 shows that the emulator could detect implausible discourses (robot in fridge). The second case (bottle is left bottle) (absurdity) is a meta-explanation. As you can see, NaivPhys4RP in autocorrect mode would have discarded it. In this case, the narrative is processed, the socio-physical graph describing the scene and resulting from the robot understanding of this context is as presented by Figure 9.3 with an explanation left and the imagination bottom. The brevity and vagueness of the narrative challenges the power of the reasoning system. Inferred graph elements are in red while others in blue. After sampling and imagining enough graphs, parallel predictions of the course of these scene particles take place as shown by Figure 9.4. Notice in the unified views of the belief that an object has a specific pose with a certain probability as in a quantum world (non-deterministic). The darker the object at a pose, the more likely is the pose. Finally, let assume that .. ***A yellow fluid is right to the large rinse fluid bottle. ... A robot is in the laboratory and this robot observes the table.*** (see medical lab project `https://www.tracebot.eu/`). The robot filters its emulations after generating and realizing them in order to recognize transparent items and estimate their 6D-poses (see Figure 9.5). **(More and HD results in video)**.

V. NaivPhys4RP Vs ChatGPT

NaivPhys4RP is not only able to process natural text like ChatGPT [11] however transparently and causally, but it is also able to transparently and causally imagine physico-realistic socio-physical scenes corresponding to those texts with no data and resource-intensive learning. While the ontology enables coarse understanding (e.g., robot not in fridge), socio-physical physico-realistic emulations enable fine-grained understanding (e.g., spoon will fall, robot will collide with door).

VI. Conclusions

In this paper, we addressed some core limitations of NaivPhys4RP while integrating the context understanding module, revising the sensor model and dynamizing the ontology. Then, we presented a first complete implementation of NaivPhys4RP for anticipating the states and observations of solid worlds as well as explaining these observations. Thirdly, we demonstrated our implementation on ultimately recognizing and estimating the 6D-Pose of objects in a safe and learningless manner from poor sensor data such as transparent objects. Finally, we demonstrated in contrast to ChatGPT how NaivPhys4RP imagines scenes from texts and demonstrates an understanding of these texts with no resource-intensive learning. As far as near future works are concerned, it is envisioned the extension of NaivPhys4RP to flexible worlds (e.g., fluids, cable) with a special attention on extending ACDL (e.g., explicit negation). Moreover, an engineering work will be carried out to make the system software exploitable by the community as much as possible. Also very crucial is the search for and establishment of benchmark environments for such extremely embodied systems.

Acknowledgment

This scientific work is partially funded by the projects DFG EASE CRC 1320 and EU TraceBot (grant agreement No 101017089).

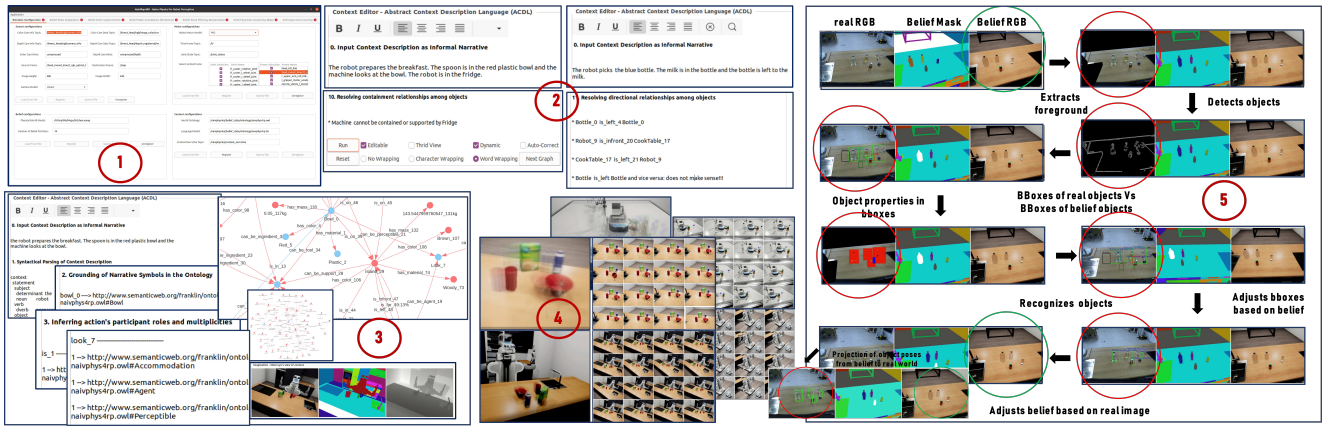

Fig. 9: Demonstrating key components on learningless and safe recognition and 6D-pose of transparent objects

REFERENCES

[1] Pieter Adriaans. “Learning as Data Compression”. In: 2007.

[2] Ruzena Bajcsy, Yiannis Aloimonos, and John Tsotsos. “Revisiting Active Perception”. In: *Autonomous Robots* 42 (Feb. 2018). DOI: 10.1007/s10514-017-9615-3.

[3] Michael Beetz et al. “Know Rob 2.0 — A 2nd Generation Knowledge Processing Framework for Cognition-Enabled Robotic Agents”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 512–519. DOI: 10.1109/ICRA.2018.8460964.

[4] Daniel Beßler et al. “Foundations of the Socio-Physical Model of Activities (SOMA) for Autonomous Robotic Agents1”. In: Dec. 2021. ISBN: 9781643682488. DOI: 10.3233/FAIA210379.

[5] Vo Duy Cong and Le Duc Hanh. “A review and performance comparison of visual servoing controls”. In: *International Journal of Intelligent Robotics and Applications* 7 (Jan. 2023). DOI: 10.1007/s41315-023-00270-6.

[6] Franklin K. Kenghagho et al. “NaivPhys4RP - Towards Human-like Robot Perception “Physical Reasoning based on Embodied Probabilistic Simulation””. In: *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*. 2022, pp. 815–822. DOI: 10.1109/Humanoids53995.2022.10000153.

[7] Franklin Kenghagho Kenfack et al. “RobotVQA — A Scene-Graph- and Deep-Learning-based Visual Question Answering System for Robot Manipulation”. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2020, pp. 9667–9674. DOI: 10.1109/IROS45743.2020.9341186.

[8] Klemen Kotar and Roozbeh Mottaghi. “Interactron: Embodied Adaptive Object Detection”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 14840–14849. URL: https://api.semanticscholar.org/CorpusID:246442335.

[9] Claudio Paolucci. *Cognitive Semiotics: Integrating Signs, Minds, Meaning and Cognition*. Springer Verlag, 2021, pp. 149–155.

[10] Thomas Parr et al. “Generative Models for Active Vision”. In: *Frontiers in Neurorobotics* 15 (2021). ISSN: 1662-5218. DOI: 10.3389/fnbot.2021.651432. URL: https://www.frontiersin.org/articles/10.3389/fnbot.2021.651432.

[11] Partha Pratim Ray. “ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope”. In: *Internet of Things and Cyber-Physical Systems* 3 (2023), pp. 121–154. ISSN: 2667-3452. DOI: https://doi.org/10.1016/j.iotcps.2023.04.003. URL: https://www.sciencedirect.com/science/article/pii/S266734522300024X.

[12] Jianwei Yang et al. “Embodied Amodal Recognition: Learning to Move to Perceive Objects”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 2040–2050. DOI: 10.1109/ICCV.2019.00213.

[13] Rui Zeng et al. “View planning in robot active vision: A survey of systems, algorithms, and applications”. In: *Computational Visual Media* 6 (2020), pp. 225–245. URL: https://api.semanticscholar.org/CorpusID:220888736.

[14] Yixin Zhu et al. “Dark, Beyond Deep: A Paradigm Shift to Cognitive AI with Humanlike Common Sense”. In: *Engineering* (2020).