Context-Independent Task Knowledge for Neurosymbolic Reasoning in Cognitive Robotics

Nicholas Hubert Kirk

Submitted by Nicholas Hubert Kirk to the University of Exeter as a thesis for the degree of Master of Philosophy by Publication in Computer Science, August 2018.

I certify that all material in this thesis which is not my own work has been identified and that no material has previously been submitted and approved for the award of a degree by this or any other University.

1st Supervisor: Dr. Nicolas Pugeault 2nd Supervisor: Dr. Chunbo Luo

Abstract

One of the current main goals of artificial intelligence and robotics research is the creation of an artificial assistant which can have flexible, human like behavior, in order to accomplish everyday tasks. A lot of what is context-independent task knowledge to the human is what enables this flexibility at multiple levels of cognition.

In this scope the author analyzes how to acquire, represent and disambiguate symbolic knowledge representing context-independent task knowledge, abstracted from multiple instances: this thesis elaborates the incurred problems, implementation constraints, current state-of-the-art practices and ultimately the solutions newly introduced in this scope.

The author specifically discusses acquisition of context-independent task knowledge from large amounts of human-written texts and their reusability in the robotics domain; the acquisition of knowledge on human musculoskeletal dependencies constraining motion which allows a better higher level representation of observed trajectories; the means of verbalization of partial contextual and instruction knowledge, increasing interaction possibilities with the human as well as contextual adaptation. All the aforementioned points are supported by evaluation in heterogeneous setups, to bring a view on how to make optimal use of statistical & symbolic applications (i.e. neurosymbolic reasoning) in cognitive robotics. This work has been performed to enable context-adaptable artificial assistants, by bringing together knowledge on what is usually regarded as context-independent task knowledge.

Contents

List of Figures

Declaration

The author hereby declares that the papers herein declared as the author's contributions have been published after a peer-reviewed process in internationally recognized conferences and workshops of the field.

Acknowledgements

My passion for artificial intelligence started as soon as I gained a firmer grasp of mathematics during my bachelor degree in Engineering. Robotics, instead, became a newly acquired taste once I approached the embodied aspect of artificial cognition. I am extremely grateful for having embarked on research in this field, which provided me insights on very diverse areas of science.

I am extremely grateful to Prof. Beetz, Prof. Cheng, and Prof. Lee for their guidance. I am indebted with my senior colleagues Daniel Nyga, Matteo Saveriano, Pietro Falco and Cristian Axenie for their practical insights and 360 degree support. I would like to thank Leif Johannsen for providing insights in psychology and in the cognitive sciences. I am grateful for the inspiring chats at conferences with Alberto Finzi and Sami Haddadin.

The content herein this thesis brings together research from different affiliations and moments: it was then united within the framework provided by the University of Exeter, and for this, I want to acknowledge the help of my supervisors Nicolas Pugeault and Chunbo Luo, for their guidance and for the pleasant talks about the present and the future.

Last but not least, I am very grateful to my parents for having supported me in this endeavor, as well as to my cousins Francesco and Andrea, my aunt Helena and my uncles Sergio and Roberto.

Nicholas H. Kirk, Exeter, U.K., August 2018

Chapter 1

Introduction

1.1 Context of the thesis

This body of work encompasses four years of work, thoughts and obsessions on the question "how to confer human context-independent task knowledge to robotic assistants". The research contained in this thesis started in Munich, Germany, both in an independent after-master effort, as well as within the "Technical University of Munich's International Graduate School of Science and Engineering" as part of the "Robotic light touch support during locomotion in balance impaired humans (ROLITOS)" project. The latter was a joint DFG-funded effort of the Chair of Human Movement Science and the Dynamic Human-Robot-Interaction for Automation Systems group of the Technical University of Munich, in which I therein investigated mathematical models of human motion representation. The environment brought together psychologists, human movement scientists, and roboticists, to provide a holistic approach to such cognitive problem. During my master and in my after-master work, I performed research on representations based on loosely-coupled inspirations of the human mind, while in the graduate school I attempted to get closer to the neuroscientific theories, such as Berstein's and Latash's. These

different perspectives, in terms of i) different level of abstraction, and ii) different degree of coupling with domain knowledge, in the form of publications, have been assembled in this thesis' scope and described by their common denominator: the generation, representation and disambiguation of symbolic knowledge, describing what is context-independent task knowledge to a human, for statistical decision making in artificial assistants (a high level overview bringing these concepts together is presented in Figure 1.1). This thesis for evaluation of the "Master of Philosophy by publication" in Computer Science at the University of Exeter is divided into chapters: i) the first chapter defines the rationale of the work and its scientific context (Chapter 1), ii) the second chapter provides an in-depth state-of-the-art analysis (Chapter 2), while the iii) third chapter aggregates the portfolio of the published work, the details of each publication venue, the details of the author's contributions, as well as a description of why the thesis constitutes a coherent body of work which satisfies the criteria set by the university (Chapter 3); iv) a final chapter provides the conclusions (Chapter 4), which comprises a set of concluding statements on the body of the portfolio of submitted work as well as future work possibilities.

1.2 Work rationale

Artificial intelligence as a field began with the intent of reproducing flexible, human-like intelligent decision making. After the creation of the first mainframes running procedural software, new paradigms were sought after for the reproduction of basic reasoning patterns. As these tasks were very knowledge intensive, the first technical focus was to reduce the workload of computers to compute already known decision making algorithms, i.e. to make these computationally feasible (Russell et al. 1995). The infeasibility was (and often is still today) laying in the excessive request of memory and execution time (Arora and Barak 2009), known as

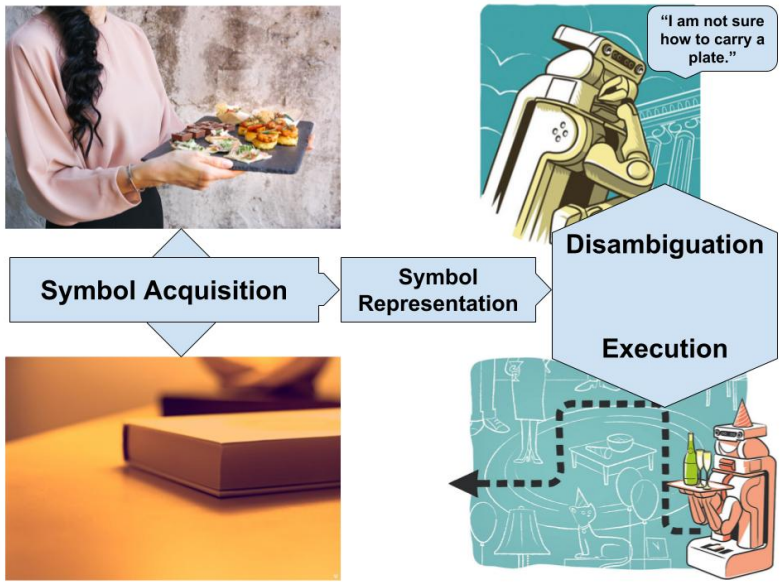


Figure 1.1: High level overview of the symbol acquisition, representation and disambiguation process presented in this thesis.

computational space and time complexity, respectively. Multiple means of reduction were therefore implemented in the form of compression (i.e. data dimensionality reduction via lossy and lossless processes) and heuristic means (i.e. implementation decisions which improve the average execution time by making assumptions on the statistically average input): these were suboptimal as they did not return accurate results nor provide guarantees for this, but they allowed to reduce average execution time. In more contemporary embedded-oriented research, a further more conceptual focus of artificial intelligence theory was and is to map percepts (i.e. inputs from sensory components) to executable actions (Russell et al. 1995), in view of service robots to autonomously acquire new skills and adapt existing ones to new tasks and environments. Robotic assistants must decide on how to perform a particular activity at runtime, which requires them to infer the appropriate actions to be executed on the appropriate objects in an appropriate way. They have to perform what is commonly referred to as *everyday activity*, which has been proven to be a very knowledge-intensive task (Anderson and Evans 1996). Timely *knowledge processing* is still a major limitation to accurate human-like decision making.

1.2.1 Why *Neurosymbolic Reasoning*

As knowledge processing is still a limitation, the acquired and represented information within cognitive frameworks of artificial assistants is in the form of *symbols*, i.e. fragments of information which are necessary and sufficient for the reproduction of an action: they capture salient information and disregard what is redundant. For this line of reasoning, it follows that it is paramount to have an extensive framework for timely acquisition, usage and disambiguation of symbols to enable various extents of embodied intelligence.

As known in literature, we make use of the term *"neuro-symbolic reasoning"* to discuss the family of technologies which enable the aforementioned symbolic format, bringing together statistical learning principles with logic formalisms: the most discussed of which is Markov Logic Networks (MLN) (Richardson and Domingos 2006), which is explained more in depth in the following pages (Section 1.3). In practical terms, this knowledge representation is able to describe well a "problem with structure" (logical structure), defining relationships among entities (such as IS-A, PART-OF), however considering also the statistical confidence attributed to such relations. This helps us to both *represent knowledge* and *reason upon such knowledge*.

1.2.2 Why *Contextual Independence*

Within this thesis scope we will discuss how our human notion of *context-independence* is in fact a criteria for how segmentation of symbols should occur, i.e. to define what information should be registered and which should be disregarded. It follows that general, abstracted knowledge is derived from multiple instance-bound contexts: the result of such knowledge induction has therefore to be context-independent (Lemaignan et al. 2010), which can later be *grounded* to a specific instance when this occurs, during the execution of an action.

1.3 Background notions

When implementing embodied intelligence to any extent, we are confronted with the question: *"what data is essential to represent a given movement, and enable its reproduction?"*. Much of such essential data is actually common to many movements and contexts, and therefore *independent from such individual contexts*. Such concepts which for humans are often of common sense in everyday activities are often referred to as the notion of *appropriateness*, in other words, the ability to execute at runtime, by deciding *the appropriate actions on the appropriate objects in the appropriate way* (Nyga and Beetz 2012). For such representation, we need to store such "appropriateness" relationships for a given action which is likely to be executed or recognized. For instance, to create a hot beverage for a human, an artificial assistant requires the knowledge of what associations are likely, unlikely and inappropriate: this in view of enabling flexibility. What would happen if one object is missing from the context, or the spatial constraints block the execution of a given action? To exemplify, consider that for instance Coffee is requested but not present in context. Tea could be the closest semantic equivalent available in context. Other possible surrogates, such as Liquid_Soap however, should be excluded from the action candidates as it is not edible. The latter exemplification shows the need for constraints on semantic representations, as well as for contextual information inducted from multiple sources. When dealing with symbol generation, representation and disambiguation, we have to ask ourselves what is an optimal policy for recreating appropriateness, and thus not relating entities which are not appropriate. The technical background to reach such goal achieves context-independence. For explanatory purposes we can define an abstract representation of a semantic relation as follows:

$$< entity_1, relationship, entity_2 >$$

Within the robotics domain, the $entity$ type can be of any abstract nature, i.e. can be an object or an action. An example can be $entity_1$:=$Apple$ and $entity_2$:=$Fruit$. The relationship is the corresponding semantic relation among the described entities, for instance, taxonomy, i.e. defining that $Apple$ is a type of $Fruit$.

What comes to mind is that representation (in terms of amount, type and logical expressiveness of the relations therein) is fundamental for the full exploitation of such semantic capabilities. How can we represent the aforementioned *appropriateness*? we require the representation of constructs which can verify, among others, that one entity is of a given class which is appropriate for a given relationship. In other words, does the instance $entity_3$:=LIQUID_SOAP satisfy the logical predicate $is\_edible(entity)$? To reply to this question, artificial intelligence has made use of more complex logical representations, one mentionable one of which is *first order logic*, which allows expression over existence and universality of complex (i.e. combined) predicates. For an in-depth understanding of this within the robotics context, the reader should refer to previous literature (Russell et al. 2003).

When talking about representation, one fundamental assumption in the applied context is that sensors have an intrinsic statistical error, which makes estimation models stochastic. It follows that logical inference based on such estimates should also take statistical error into account: hence the creation of "*soft*" inference approaches. This is performed thanks to *neurosymbolic* systems, i.e. a set of logico-statistical relations in the form of logical clauses with statistical weighting appended. The reasons for using the latter are manifold, in terms of practical expressiveness during use, one of which is the enabling uncertain inference, often useful given the partial availability of evidence of many deduction applications. In terms of representation in this scope we will discuss solely Markov Logic Networks (MLN) (Richardson and Domingos 2006), as this is a proven, widely adopted formalism, which is the basis of the majority of published papers provided within

this thesis as well as in many pieces of research in cognitive robotics (Tenorth and Beetz 2009). MLN provided weighted satisfiability and inference over first order logic clauses. More formally, in MLN a world belief is expressed as:

$$\Pr(X = x) = \frac{1}{Z} exp\left(\sum_j w_j f_j(x)\right) \tag{1.1}$$

In this (1.1), the model defines a probability over a given world x as a log-linear model in which we have an exponentiated sum of weights w_j of a binary feature f_j, and the partition function Z. For a more in depth explanation, the reader should refer to (Richardson and Domingos 2006) for the formalism definition and (Kok and Domingos 2005) for the applications.

In this thesis the author now provides first a literature-based state of the art analysis to show what has been achieved in symbolic reasoning in recent years (with regards to symbol segmentation, disambiguation and usage) (Chapter 2), to then discuss the list the contributions to the field of the submitted portfolio (Chapter 3).

Chapter 2

How the work forms a coherent whole

2.1 Contributions to neurosymbolic reasoning

We now describe the specific areas the presented portfolio of papers contributes to, by providing a brief description of the immediate background as related work, as well as an explanation of the present limitations and the author's contributions to the field.

2.1.1 Generation

To assemble a large set of semantic concepts and relations, e.g. what objects are likely related in a task execution, in what role, as well as what associations are inappropriate, we require a great amount of information. Assumptions on the likelihood of association, also non-trivial ones, can be inferred by performing frequency-based analysis on large amounts of data, specifically text. This concept,

known as *distributional hypothesis*, was first introduced in computational linguistics for semantic characterization on the basis of word co-occurrence (Harris 1954), and furthered in more abstract artificial intelligence tasks such as machine translation (Mikolov et al. 2013). When fully constructed, a semantic knowledge base enables decision making and execution on the basis of partial sensory data, or more formally described, this aids the grounding of action predicates with anchored entities (Coradeschi and Saffiotti 2003), to then translate a higher level response to specific sensorimotor commands (Wächter et al. 2013; Kraft et al. 2008; Krüger et al. 2011).

An immediate question a reader might have is how to retrieve and construct such large set of human knowledge in machine-readable format: It has been empirically proven that we can assume that some common latent semantics exist between motion observations and written words (Takano and Nakamura 2008). It is therefore fundamental to exploit the large sources of notions such as long texts, minimizing manual annotation work made by the human. *Limitations of the current approaches:* As per generation of concept representations based on text mining, the described systems have not been used within the robotics context.

The author's contribution: the use of text as source of frequency of co-existence analysis of abstract entities, within the robotics context, is novel. The work focuses on how to achieve scalable affordance mining from human-written texts, where an affordance is the set of possible actions an object can be subject to, or that an object can actively execute (for example, an apple has passive affordances such as is_edible and $is\_boilable$, while any human has active affordances such as can_eat and can_boil). The presented contributions from the author (Kirk 2014, Bhalla and Kirk 2016) provide a prototype and an evaluation of vector space semantics for the use of affordance mining for use in robotics. These semantic spaces can be used as a knowledge base (KB) when inferring the likelihood of in-

coming actions, given contextual information. In the author's view, constructing a large affordance database is the first cornerstone to achieve context-independent reasoning on everyday activities.

2.1.2 Representation

In terms of symbol representation, past work has focused on dimensionality reduction, such as neural auto-encoding approaches on numerical values (Hinton and Salakhutdinov 2006), or other symbol-level attempts (Kaltenbacher et al. 2015, Cangelosi et al. 2000). Other mentionable work has instead focused on less compact representations which enable expressive (first-order logic level) logical inference (Richardson and Domingos 2006, Sutton and McCallum 2006), thereby defining an optimal relational formalism for *problems with structure*. This work has enabled well engineered research infrastructures with real-time application capabilities in embodied contexts (Tenorth and Beetz 2009; Beetz et al. 2015a; Beetz et al. 2015b), often enabling interaction with the human, both linguistically and physically (Cangelosi et al. 2006). When debating more trajectory-level representations, symbols are less abstract in nature as they have little to no parametrization, and are discussed as motion *primitives*. These have been widely debated with respect to usage of various latent estimation methods (Kulic et al. 2008; Kulic et al. 2009). Other work focuses on partial body representations of human movement (Ficuciello et al. 2018).

Limitations of the current approaches: With regards to motor primitives, past work has focused on the extent to which different latent parameter estimations (such as Hidden Markov Model estimation) can classify actions based on training performed on a diverse data set, but provided excessive focus on empirical-based model estimates rather than on sound assumptions of human movement.

The author's contribution: A dimensionally compact representation of movement, based on human motor coordination principles, has been developed and evaluated for the purpose of action recognition and representation (Falco et al. 2017). The scientific introduction by the author is to exploit the notion, well-known in neuromechanics (Latash et al. 2007), that humans move their joints in a coordinated fashion. The neuromechanical community, in fact, has widely accepted that human movement is a coordination of muskulo-skeletal features with various degrees of intentional stability and control (Latash 2010). The highly influential theories of motor primitives and movement synergies assume that the various degrees of freedom of a motor system are not controlled independently, but instead present couplings and dependencies. This is also known in the Leading Joint Hypothesis (LJH) (Dounskaia 2005). One of the main contributions of the author's contribution is to exploit such correlation among joints to increase the performance of action recognition in terms of accuracy and scalability. The presented contribution, in addition to the high level classification rate on well known datasets of movement such as HDM05 (Müller et al. 2007), provides a representation of movement with very low computational complexity, useful for practical real-time applications of movement representation or recognition. The novelty mainly lies in the usage of neuroscientifically sound assumptions, namely the motor couplings inherently used by the central nervous system of humans. The author, externally from this thesis, also contributed to the community with a formalism at a higher level of abstraction, making use of statistical relational learning, which combines both semantic memory and sequence modeling for more accurate prediction of actions (Kirk et al. 2015).

2.1.3 Disambiguation

In this scope we identify the *disambiguation* problem as the issue arising when there is the need to identify the action to be performed from partially known instruction data. The disambiguation problem will always be present, given: 1) the always probabilistic nature of sensory perceptions, and 2) the intrinsically underspecified nature of natural language instructions provided by humans (Nyga and Beetz 2012). For this, research has focused on mathematical inference possibilities to improve decision making estimates (Nyga and Beetz 2012, Magnanimo et al. 2014).

Limitations of the current approaches: While some work has focused on human-robot interaction for intention disambiguation on a visual-motor level (Saveriano 2017), and other has worked with language association (Cangelosi et al. 2006, Nyga and Beetz 2012), previous work lacks linguistic disambiguation, i.e. the possibility of the robotic assistant to reply to the human asking specific fragments of information which are not available with a sufficiently high confidence interval.

The author's contribution: In the presented research (Kirk et al. 2014), together with a mean for verbalizing doubts, where doubts are intended as fragments of information which are to some degree uncertain, the author's work provides an output in both a machine- and human-readable format known in computational linguistics, i.e. Controlled Natural Languages (CNL). Such verbalization formalization is compatible with both first order logic inference and natural language interaction, allowing in the contributed research the implementation of "turn taking" of questions & answers between the robotic assistant and the human, for disambiguation of non-inferrable, underspecified natural language instructions. Verbal disambiguation of symbols is a fairly unexplored area of human-robot interaction (Thomaz et al. 2016). While already proven as powerful formal-

ism of representation and reasoning for the semantic web (Schwitter and Tilbrook 2004; Schwitter 2010), the presented claim is that novel uses of CNL are possible for robotic assistants, specifically as robot-human interface.

Chapter 3

Author's Contributions

3.1 Portfolio

The portfolio submitted for consideration of the degree of MPhil by publication comprises the following publications:

- **Kirk, Nicholas Hubert, Daniel Nyga, and Michael Beetz. 2014. 'Controlled Natural Languages for Language Generation in Artificial Cognition.' In 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 6667–6672.**
  Authorship statement: The author, given an initial problem statement and given supervision, performed all research, implementation and validation, and wrote the majority of the paper.

- **Falco, Pietro, Matteo Saveriano, Eka Gibran Hasany, Nicholas Hubert Kirk, and Dongheui Lee. 2017. 'A Human Action Descriptor Based on Motion Coordination.' In IEEE Robotics and Automation Letters 2 (2): 811–818.**
  Authorship statement: The author contributed to the paper specifically by i) suggesting the exploitation of joint variance in this context, ii) suggesting the

exploitation of joint correlation in this context, iii) researching the relevant human movement science literature (e.g. LJH, Bernstein, Latash) to justify the neuromechanical hypothesis, and iv) co-wrote minor parts of the text.

- **Kirk, Nicholas Hubert. 2014. 'Towards Learning Object Affordance Priors from Technical Texts.' In "Active Learning in Robotics" Workshop, 2014 IEEE-RAS International Conference on Humanoid Robots.**

- **Bhalla, Vishal A, and Nicholas Hubert Kirk. 2016. 'Prior Affordance Understanding with Relational Learning for Human Safe Action Planning.' In "AI for Long-Term Autonomy" Workshop, 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden.**
  Authorship statement: The author made a first joint authorship contribution, i) providing the initial idea, ii) providing the research boundaries and hypothesis definition, as well as iii) giving the definition of the experiment for validation and iv) co-writing the text.

The submitted research intends to highlight the importance of symbolic information for achieving context-awareness and representation: semantic information allows for generalization of multiple different instances, allowing context abstraction. Particular focus is on i) the retrieval of symbols from heterogeneous sources (such as motion trajectories (Falco et al. 2017), or text corpora (Bhalla and Kirk 2016)), ii) the suitable mathematical and human-like representation (and the consequences in terms of tractability of the algorithms which perform inference operations on such symbols) (Kirk et al. 2014), and iii) the disambiguation of symbolic information within a human-robot interaction setting (Kirk et al. 2014).

3.2 Fulfillment of the MPhil assessment criteria

To provide a reply to the fulfillment of the university conditions stated in "5.1 Regulations Governing Academic Programmes "[1], the author states that the current submitted portfolio of work has been internationally peer-reviewed in IEEE conferences and workshops. Precisely, (Kirk et al. 2014) and (Falco et al. 2017) are considered to be the highest level of conference impact within the field (specifically ICRA), while (Bhalla and Kirk 2016) and (Kirk 2014) have been presented during workshops in the aforementioned ICRA conference and in the HUMANOIDS conference: this constitutes proof of extension of knowledge.

1. https://www.exeter.ac.uk/staff/policies/calendar/part1/regulations/r2-1/ - last accessed 28th August 2018

Controlled Natural Languages for Language Generation in Artificial Cognition

Nicholas H. Kirk[1], Daniel Nyga[1,2], Michael Beetz[2]
[1]Intelligent Autonomous Systems, Technische Universität München, Germany
[2]Institute for Artificial Intelligence & TZI, University of Bremen, Germany
nicholas.kirk@tum.de, nyga@cs.tum.edu, beetz@cs.uni-bremen.de

***Abstract*— In this paper we discuss, within the context of artificial assistants performing everyday activities, a resolution method to disambiguate missing or not satisfactorily inferred action-specific information via explicit clarification. While arguing the lack of preexisting robot to human linguistic interaction methods, we introduce a novel use of Controlled Natural Languages (CNL) as means of output language and sentence construction for doubt verbalization. We additionally provide implemented working scenarios, state future possibilities and problems related to verbalization of technical cognition when making use of Controlled Natural Languages.**

I. Introduction

In everyday routine activities, robotic assistants and co-workers will have to perform a variety of tasks for which they cannot be pre-programmed because they are not known at production time. Research in the field of cognitive robotics envisions service robots that autonomously acquire new skills and adapt existing ones to new tasks and environments. They must decide on *how* to perform a particular activity at runtime, which requires them to infer the *appropriate* actions to be executed on the *appropriate* objects in an *appropriate* way. They have to perform what is commonly referred to as *everyday activity*, which has been proven to be a very knowledge-intensive task [1], [2]. It requires context awareness and flexibility in action parametrization when operating in real world settings with partially available information, which is referred to as the "open world challenge".

Recent research in the field of cognitive robotics aims to make knowledge sources available for robots, which have been created by humans and are intended for human use. For some domains such as daily household tasks (e.g. cooking, cleaning up), step-by-step plans and recipes from web pages like *wikihow.com* have been successfully used for feeding such common sense knowledge about actions and objects into knowledge bases of mobile robotic platforms and for transforming such recipes into executable robot plans [3].

However, as these recipes are presented in natural language, severe ambiguity, vagueness and underspecification have been identified as major challenges in translating such specifications into plans, since many missing key pieces of information are generally considered common sense to the human. As an example, consider the natural-language instruction "Flip the pancake", taken from a recipe for making pancakes: In order to perform the action successfully, a robot needs for instance to decide which utensil to use

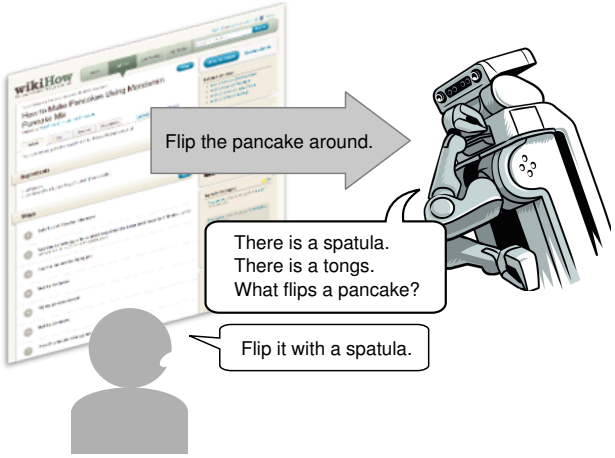


Fig. 1. Representation of a disambiguation interaction

(e.g. a spatula), where to hold it (e.g. at its handle) and what part of it to put underneath the pancake (e.g. the blade), and where to flip it from (e.g. the burner). Current research in cognitive robotics [4] aims to build action verb-specific knowledge bases that fill these knowledge gaps and enables a robot to infer the information which is *needed* in order to perform a particular activity, based on what is *given* in a naturalistic action specification.

However, such inference might be insufficient to formulate action specification and require the robot to fall back on human assistance. As an example, consider a situation where the robot is asked to flip a pancake, but the knowledge base does not contain sufficient information about what instrument is to be used (e.g. it has no strong preference for a spatula over barbecue tongs). In such a case, the robot has to explicitly ask a human for instrument clarification. Fig. 1 illustrates such a situation. Taxonomical and compositional relationships or object role understanding are only some of the missing elements that could potentially require clarification.

In this work, we present an implementation of a novel approach to autonomously identify and verbalize such absent information in a knowledge base, in order to enable a robot to actively enhance its knowledge about actions and objects by stepping into dialog with humans.

The contribution of this paper, within the artificial cogni-

tion domain, is the making use of *CNL as mean of language generation*: we use CNL as output language of our doubt verbalization procedure, where doubt is intended as the non autonomously removable uncertainty related to objects involved in the action. Such situation then requires human intervention for appropriate translation to action plans. Our contribution lies in the conceptualization and implementation of the doubt case classification, the discourse abstraction of the robot reply and the verbalization procedure of the latter.

The remainder of this paper presents a description of the adopted technologies (*PRAC, ACE, DRS*); an explanation of how the claims are related to the state of the art; an explanation of the implemented language generation module; a system evaluation and an ending summary comprising results, current limitations and future perspectives.

II. Adopted Technologies

Before explaining the details of the dialog-based disambiguation, we now describe the adopted technologies we make use of as source of inferred information (PRAC), and the human-readable target formalization (ACE), used also as support to the verbalization procedure itself (ACE and DRS).

A. Probabilistic Robot Action Cores

Nyga et al. [2] introduced the concept of *Probabilistic Robot Action Cores* (PRAC), which can be thought as abstract, generic event patterns representing sets of inter- and intra- conceptual relations that constitute an abstract event type, assigning an action role to each entity that is affected by the respective action verb. Formally, a PRAC is defined as a conditional probability distribution:

$$P\left(\mathcal{R} \times \mathcal{A} \times \mathcal{C} \mid \sqsubseteq, \preceq\right) \quad . \tag{1}$$

$\mathcal{R}$ is the set of all action roles
$\mathcal{A}$ is the set of all action verbs
$\mathcal{C}$ is the set of all class concepts
$\sqsubseteq$ is a taxonomy relation over $\mathcal{C}$
$\preceq$ is a mereological relation over $\mathcal{C}$

For terminology explanations we refer to [2]. As opposed to most approaches towards understanding natural-language instructions, which merely seek to understand what is given by an instruction [5] [6], the PRAC concept is also able to infer what is missing. It combines action-specific and ontological world knowledge in a joint probabilistic first-order representation, which allows to automatically find generalizations from concrete event occurrences at an appropriate level of abstraction. Specifically, PRAC models are represented as a set of action roles (action parameters defining relations among entities involved in an action) and Markov Logic Networks (MLN), a knowledge representation formalism that combines first-order logic and probability theory [7]. Fig. 2 provides an example of the PRAC model for the action core '*flipping*'.

- *Action Core:* Flipping
- *Definition:* An Agent causes a Theme to move with respect to a FixedLocation, generally with a certain Periodicity, without undergoing unbounded translational motion or significant alteration of configuration/shape.
- *Action Roles:*
  - *Theme*: A physical entity that is participating in non-translational motion.
  - *Instrument*: An entity that is used to perform the flipping action.

Fig. 2. "Flipping" Action Core, and an enumeration of its Action Role definitions (partially adopted from *FrameNet* [8]). MLN formulas are not listed for readability.

A PRAC defines a joint probability distribution over the action roles according to Eq. 1, such that arbitrary parameter slots, which are not given in an NL instruction, can be inferred based on what has been stated explicitly in such instruction.

B. Attempto Controlled English & Discourse Representation Structures

Fuchs et al. [9] presented Attempto Controlled English (ACE), a general purpose Controlled Natural Language (CNL), i.e. a subset of standard English with restricted lexicon, syntax and semantics, formally described by a small set of construction rules and a controlled vocabulary. This allows a text in CNL to be read naturally by any person that knows the natural language it stemmed from, even if unaware of the underlying formalizations, and is more readable than other traditional formal languages [10].

Being a formal language, CNL can be proved by automatic theorem proving software, translated into First Order Logic or OWL ontology representations and also be paraphrased into paratactic noun sentences. ACE provides linguistic constructs that are usually present in natural languages, such as countable nouns (e.g. 'robot', 'pancake'), proper names ('John'); universal, existential, generalized quantifiers ('all','a','at least 2'); indefinite pronouns ('somebody'); intransitive, mono- and di-transitive verbs ('sleep', 'like', 'give'); anaphoric references to noun phrases through definite noun phrases and pronouns; composite sentences as compounds of coordination, subordination, quantification, and negation phrases. Fig. 3 provides an example of some of such constructs in an ACE sentence, and also the 'paraphrased understanding', i.e. the breakdown of the latter into paratactic noun phrases, making use of cross-sentence references (i.e. X1, X2 in the example).

ACE: "A robot who does not understand asks a human that knows."

There is a robot X1. There is a human X2. The human X2 knows. The robot X1 asks the human X2. It is false that the robot X1 understands.

Fig. 3. example sentence in Attempto Controlled English, followed by the 'paraphrased understanding' of such sentence via the use of ACE parser

These paraphrase-obtained noun phrases have a two-way relationship with Discourse Representation Structures (DRS) [11], a format to encode information of multiple sentences, preserving anaphoric references (i.e. *discourse referents*). Fig. 4 provides an example of cross-sentence referencing and universal quantification within the DRS formalism.

"Every robot is made by a human."

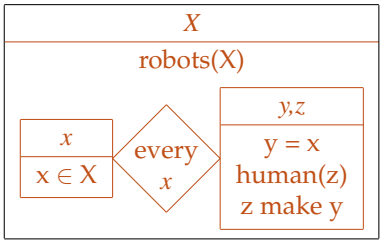


Fig. 4. explanatory ACE sentence & related DRS example describing cross-sentence references and universal quantification

III. Related Work

In reference to filling missing information such as objects or actions in verb-oriented formalizations, a large amount of research has been done in order to provide databases of conceptualizations of actions [8], [12], [13]. However, these projects do not provide computational models for inference and learning, and they do not address the problem of autonomously identifying and closing such gaps of knowledge.

Regarding verbalization, ACE recently has been exploited for uses such as Semantic Web Ontologies [14] and Multilingual semantic wikis [15], while this paper focuses on the contributions of ACE in the artificial cognition domain. Formalisms in artificial cognition oriented towards natural language understanding do exploit grounding of words to abstract objects [16], but do not comprise means of verbal interaction for disambiguation purposes. According to what is known to the authors to date, the ACE verbalization functionality itself [14] has been used uniquely for verbalizing OWL ontologies. While using the same means (i.e. ACE and the DRS verbalizer, the latter being an intermediate phase of the OWL-to-ACE verbalizer) we exploit the system for verbalizing questions and ambiguity statements, in order to verbalize a probabilistic knowledge formalism.

IV. CNL for Task Querying

Given a Natural Language Instruction (NLI), the PRAC system caters for action verb and roles understanding, inferring missing candidate objects involved in the action. Unfortunately, this operation can be partially satisfactory and some residual doubt might require explicit verbal clarification, in order to avoid that partially inferred information is translated into action planning.

Fig. 5 represents the interaction that can occur until the robotic assistant reaches a sufficient level of understanding. Taxonomical, temporal, substitution, impossibility clarifications are only some of the *disambiguation cases* in which an explicit verbal task querying is necessary from the robot

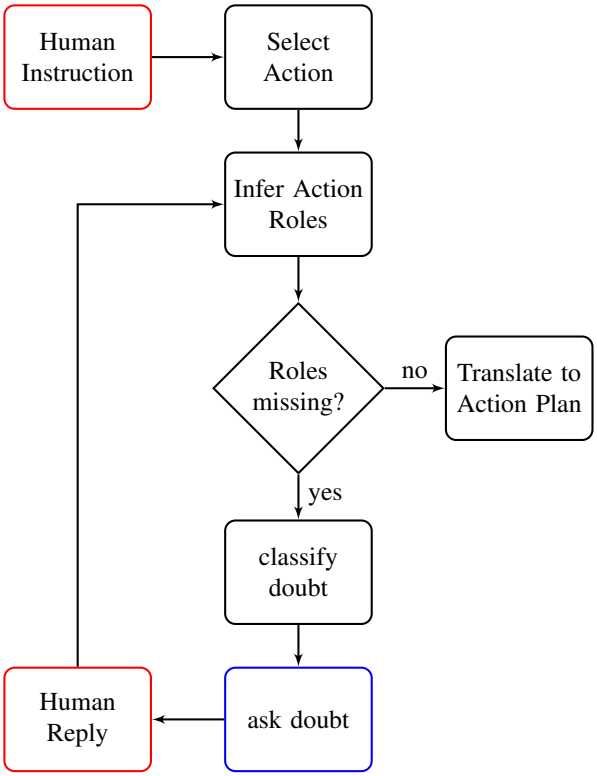


Fig. 5. A high-level flow chart of the dialog-based action role disambiguation procedure.

to the human. The generation of natural-like language is non-trivial given the scalability issues of linguistic factors of the sentence construction (e.g. anaphora resolution, number/gender/particle concordance, subordinate sentence handling, punctuation, verb conjugation). The latter requirements have proven to be fulfilled by Attempto Controlled English (ACE), now used as means of sentence construction. To do so, we make use of the ACE system's verbalization functions, used to generate ACE sentences from ontological knowledge. Specifically for an action-oriented representation, we present our cognitive verbalization procedure, operated for each non-assigned action role:

1) identification of the *disambiguation cases* (i.e. type of doubt)
2) articulation of such doubt in a statement encapsulating what was inferred, and an interrogative sentence
3) integration of the reply, assigning the previously missing action roles

A. Implementation

We now describe the implementation of the system by defining a pseudocode (Algorithm 1) that incorporates references for the sub-functions hereafter described (i.e. C0,C1,C2,C3,C4,C5).

a) action role inference (C0,C1): C0 operates an instantiation of the action template slots with the most likely class concepts, derived from the joint probability distribution of Eq. 1, while C1 retrieves the full enumeration of slots for

Algorithm 1: Action Role Disambiguation via H-R dialog

Data:
- $PRAC$, the joint probability distribution over all used class concepts given ontology knowledge, formally $P\left(\mathcal{R} \times \mathcal{A} \times \mathcal{C} \mid \sqsubseteq, \preceq\right)$.
- $ActionDB$, a template database with all ActionRoles for each ActionVerb

Result: satisfactory assignment of all the ActionRoles required by the template for successful translation into action planning.

begin

wait for NLI

C0 $K \longleftarrow pracInference(NLI)$

$t \longleftarrow$ sentence verb from K

C1 $T \longleftarrow retrieveTemplateRoles(t, ActionDB)$

$U \longleftarrow T \setminus K$

while $U \neq \emptyset$ **do**

remove item u from list of U with minimum syntactic relationship arity (known in template)

C2 $doubtCase \longleftarrow doubtCaseIdentification(u)$

C3 $queryType \longleftarrow$ retrieve the grammatical type related to the missing role u

$sDrsT \longleftarrow pullDrsCase(doubtCase)$

$qDrsT \longleftarrow pullDrsCase(queryType)$

$sParam$, $qParam \longleftarrow$ inferred or known contextual information necessary for the grounding of specific templates of statement and question

$sDrsGnd$,$qDrsGnd \longleftarrow$ grounding of $sDrsT$ and $qDrsT$, via syntactic substitution of $sParam$ and $qParam$, respectively.

C4 $aceS \longleftarrow verbalizeDRStoACE(sDrsGnd)$

$aceQ \longleftarrow verbalizeDRStoACE(qDrsGnd)$

output aceS

output aceQ

wait for $reply$

C5 $N \longleftarrow pracInference(reply)$

$K \longleftarrow K \cup N$

$U \longleftarrow T \setminus K$

comparison reasons. A lack of assignment to an action role by C0 can be due to the impossibility of defining a likely candidate (all probability assignments are below a threshold), the presence of manifold candidates (probability assignments are too close), or the optimal candidate is not available in context. For a more formal and in-depth description of such process we refer to [2].

b) case identification (C2): is operated when a role slot stated in our action core template has not yet been assigned for the previously described reasons. Case identification is performed via threshold evaluation of probability values of the most likely concept candidates for the missing role.

More formally, let $fstLikely$ be:

$$\arg\max P(neededRole \mid knownRoles, KB\left(\sqsubseteq, \preceq\right)). \quad (2)$$

We then can describe our selection procedure as:

if $P\left(fstLikely\right) < possibilityThreshold$ **then**
  return "Impossibility"
if $P\left(fstLikely\right) < P\left(sndLikely\right) - proxThreshold$ **then**
  return "Two-choices"
⋮
return "None"

Such abstract cases are represented in Discourse Representation Structure (DRS) templates that also comprise explanatory information.

TWO-CHOICES:
query_case: doubt between two plausible objects
ACE_template: There is a X1, there is a X2.
drs: drs([A,B],[object(A,X1,countable,na,eq,1)-1/4, object(B,X2,countable,na,eq,1)-1/9])
dependencies: PARAM1-ext, PARAM2-ext; X1, X2

Fig. 6. DRS template of a twofold doubt choice for a role assignment

Fig. 6 provides an example for such template. The explanation of the various fields is the following:

- *query_case* is a high-level descriptive sentence of the case
- *ACE_template* present only for explanatory reasons, is a sentence in ACE that represents, still in a template form, what the output would look like after verbalization
- *drs* is the uninstantiated discourse representation of *ACE_template*: the markers (in the example, X1, X2) will be syntactically substituted with PRAC inferred information upon template grounding
- *dependencies* defines for retrieval and substitution purposes, the type of parameters that are needed to perform grounding, and their corresponding marker in the template. Such parameters have syntactic relationships with other roles involved in the action (described in C3)

c) typed dependency parsing (C3): Together with a statement of the doubt case identification that encapsulates contextual information, a specific object query will also be verbalized in the form of a question. All template action roles have a 2-way relationship with a grammatical type within the scope of the action. These types are abstracted in DRS cases (same as to the DRS modeling described in C2, e.g. in Fig. 8), that need to be retrievable given the missing action role.

The knowledge regarding the association between the action roles and the grammatical type is provided by a *controlled template*, an ACE sentence that comprises all uninstantiated action roles in a possible syntactic configuration, built upon PRAC template model construction (an abstraction

for all instances of that action verb, e.g. 'flipping'). Fig. 7 provides an example of such modeling.

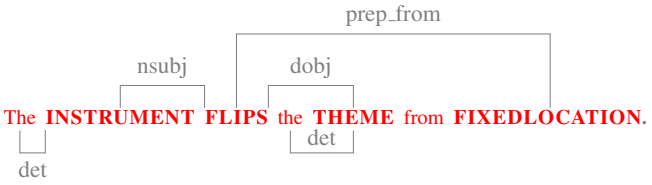


Fig. 7. *Controlled template* example for ActionVerb '*flipping*' (in red), and the typed dependencies among the words of such sentence (in grey)

The grammatical relationships in such a CNL sentence provide a formal understanding of the language-explicit relationships between the entities involved in the action. The relationships and the type of the dependencies from the latter are obtained by processing the sentence with a typed dependency parser (for our implementation, the Stanford Parser [17], for which we also refer to for type clarification).

INSTRUMENTAL:
query_case: preposition of instrument
template: What X1 the X2?
drs: drs([],[question(drs([A,B,C],[query(A,what)-1/1, object(C,X2,countable,na,eq,1)-1/4, predicate(B,X1,A,C)-1/2]))])
dependencies: NSUBJ-left, NSUBJ-right; X1, X2

Fig. 8. DRS template of an 'instrument' object query

d) sentence construction (C4): is to provide a grammatical structure for the grounded discourse abstractions of the case identification statement and the object query question. This is implemented by using the ACE verbalizer functions, providing grounded DRS instances as a formal parameter.

The approach of verbalization of two phrases, namely doubt statement and object question, has been chosen to provide the human with a better understanding of both the uncertainty (e.g. what instrument should be used) and of what has been inferred (e.g. spatula and tongs are the most likely candidates).

e) reply integration (C5): is performed by making use of the previously described action role inference routine on the natural language reply. After retrieving the new action role assignments, we will substitute in the main instance only the newly identified action roles that were previously missing. The pipeline of reasoning is shown with an example in Fig. 9.

V. Evaluation

As performance measures we take into consideration the natural likeness, the morphosyntactic correctness and the ability to convey the wanted meaning of the CNL output of our verbalizer system.

We operated our evaluation based on two action cores (i.e. Flipping, Filling) that comprised full trained models for

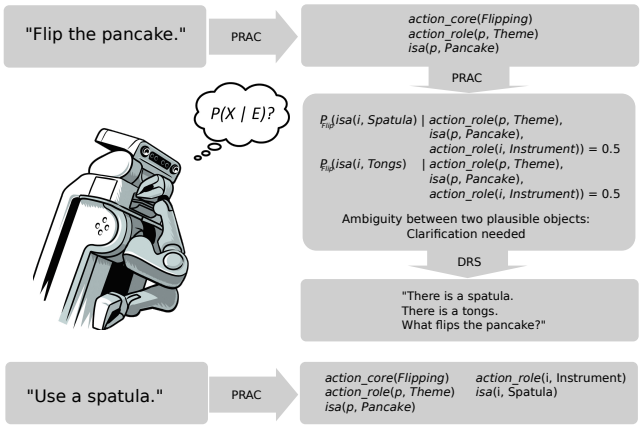


Fig. 9. Reasoning pipeline of a possible instance of disambiguation interaction

PRAC inference in order to verify full pipeline evaluation (example in Fig. 9), and we made use of various other arbitrary NLI sentences to verify correct typed dependency parsing, assignment and grounding of the doubt discourse representations. For the tested domain, the disambiguation verbalization outputs were intelligible and conveyed the meaning, but were not perfectly natural given the lack of the use of modal verbs (i.e. "what flips the pancake?" instead of "what can flip the pancake?"). An evaluated test instance and the related pipeline information is illustrated in Fig. 9.

Given that the verbalizer system is purely a PRAC and DRS based syntactical manipulator, we can assume scalability of our system within the running assumptions and performance bounds of the underlying systems [2] [18].

According to our evaluation, orthography of wording remains intact from NLI to ACE output (the latter partially exploits the same wording), as long as the verbalizer makes use of words that are part of the running ACE system's vocabulary (that can be modified dynamically), otherwise the ACE system will add explicit syntactic tags to highlight the nature of the Part-Of-Speech of such words.

No difference has been observed when making use of action cores based on intransitive, mono- and di-transitive verbs, since the DRS disambiguation cases parameters target typed dependencies that differentiate among direct and indirect objects [17]. Regarding the natural likeness of such sentences, readability studies of ACE have already been undertaken [10].

Regarding the potential ability of conveying the meaning of the doubt given the PRAC abstraction, it is up to who manually constructs the DRS templates, aligned to an ACE output, to be able to exploit the expressiveness of the ACE rules, and will also be constrained by the latter.

VI. Other uses of CNL

Future work can exploit CNL in the artificial cognition context differently, namely by using CNL as serialization of the action oriented formalism, but with the use of semantically unambiguous nouns, therefore with fully deterministic denotations.

Specifically for the formalization of a PRAC model, be it grounded or abstract, we require a format that can serialize an instance or generate the PRAC model template: that will comprise all generative information, namely all roles involved in the action, and data that can create the MLN formulas that evaluate the probability of all possible grammatical types the action roles can be involved in.

We hypothesize that this can be achieved by defining the action roles as nouns in a Controlled Natural Language statement, for human-readability and for preserving grammatical relationships between objects; furthermore we add explicit semantic tags to maintain information regarding semantic disambiguation of objects. An implementation is potentially possible via the use of ACE and semantic tags from WordNet [19]. Fig. 10 provides an example. Any proof of concept of such hypothesis is left as future work.

The AGENT.n.06 FLIPS.v.08 the THEME.n.01 from LOCATION.n.01 with an INSTRUMENT.n.01.

Fig. 10. example of a Controlled Natural Language statement, comprising nouns with semantic tags for action-oriented formalism serialization

VII. Results, Discussions and Conclusions

This paper brings attention to possible uses of Controlled Natural Languages (CNL) in the artificial cognition domain. While already proven as powerful formalism of representation and reasoning for the semantic web [14], our claim is that novel uses of CNL are possible for robotic assistants, specifically as robot-human interface. We have proven via a formalization and a practical implementation that CNL can be exploited as means for sentence construction and target language of verbalization procedures.

However, even if discourse representation is an easier instrument for achieving knowledge engineering, CNL construction is not always straightforward [20]. In fact, the DRS construction of the disambiguation cases has to account for the ACE construction rules (that can present expressiveness limitations) and the asymmetry of what is accepted as correct ACE statement and what can be verbalized (e.g. modals). The verbalizer system, being purely a DRS and PRAC based syntactical manipulator, is constrained by the current implemented features of these and presents similar limitations. This is visible since the verbalization outputs are readable but not perfectly natural-like sentences, and can present scalability issues given by improper PRAC object inference. With the expansion of the expressiveness set of ACE and DRS, future work will aim towards understanding how to make use of such abstractions in order to provide robotic assistants with more language constructs and modalities of speech. Further research will be dedicated to the consolidation of the presented proof of concept, and will focus on the interaction dialogue in order to enable further *learning via human-robot verbal interaction* capabilities.

Acknowledgments

This work has been partially supported by the EU FP7 Projects *RoboHow* (grant number 288533) and *ACAT* (grant number 600578).

References

[1] J. E. Anderson, "Constraint-directed improvisation for everyday activities," Ph.D. dissertation, University of Manitoba, 1995.

[2] D. Nyga and M. Beetz, "Everything robots always wanted to know about housework (but were afraid to ask)," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vilamoura, Portugal, October, 7–12 2012.

[3] M. Tenorth, D. Nyga, and M. Beetz, "Understanding and executing instructions for everyday manipulation tasks from the world wide web," in *IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, AK, USA, May 3–8 2010, pp. 1486–1491.

[4] M. Tenorth and M. Beetz, "Knowrobknowledge processing for autonomous personal robots," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 4261–4266.

[5] C. Matuszek, D. Fox, and K. Koscher, "Following directions using statistical machine translation," in *Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2010, pp. 251–258.

[6] S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. J. Teller, and N. Roy, "Understanding natural language commands for robotic navigation and mobile manipulation." in *AAAI*, 2011.

[7] M. Richardson and P. Domingos, "Markov logic networks," *Machine Learning*, vol. 62, no. 1-2, pp. 107–136, 2006.

[8] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The berkeley framenet project," in *Proceedings of the 17th international conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, 1998, pp. 86–90.

[9] N. E. Fuchs, K. Kaljurand, and G. Schneider, "Attempto Controlled English Meets the Challenges of Knowledge Representation, Reasoning, Interoperability and User Interfaces," in *FLAIRS 2006*, 2006.

[10] T. Kuhn, "An evaluation framework for controlled natural languages," in *Proceedings of the Workshop on Controlled Natural Language (CNL 2009)*, ser. Lecture Notes in Computer Science, N. E. Fuchs, Ed., vol. 5972. Berlin / Heidelberg, Germany: Springer, 2010, pp. 1–20.

[11] H. Kamp and U. Reyle, *From discourse to logic: Introduction to model theoretic semantics of natural language, formal logic and discourse representation theory*. Kluwer Academic, 1993, vol. 42.

[12] K. K. Schuler, "Verbnet: A broad-coverage, comprehensive verb lexicon," 2005.

[13] P. Kingsbury and M. Palmer, "From treebank to propbank," in *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*. Citeseer, 2002, pp. 1989–1993.

[14] K. Kaljurand, "Attempto Controlled English as a Semantic Web Language," Ph.D. dissertation, Faculty of Mathematics and Computer Science, University of Tartu, 2007.

[15] K. Kaljurand and T. Kuhn, "A multilingual semantic wiki based on Attempto Controlled English and Grammatical Framework," in *Proceedings of the 10th Extended Semantic Web Conference (ESWC 2013)*. Springer, 2013.

[16] S. Lemaignan, R. Ros, E. A. Sisbot, R. Alami, and M. Beetz, "Grounding the interaction: Anchoring situated discourse in everyday human-robot interaction," *International Journal of Social Robotics*, vol. 4, no. 2, pp. 181–199, 2012.

[17] M.-C. de Marneffe and C. D. Manning, "The stanford typed dependencies representation," in *COLING Workshop on Cross-framework and Cross-domain Parser Evaluation*, 2008. [Online]. Available: pubs/dependencies-coling08.pdf

[18] N. E. Fuchs, K. Kaljurand, and T. Kuhn, "Discourse Representation Structures for ACE 6.6," Department of Informatics, University of Zurich, Zurich, Switzerland, Tech. Rep. ifi-2010.0010, 2010.

[19] G. A. Miller, "Wordnet: A lexical database for english," *Communications of the ACM*, vol. 38, pp. 39–41, 1995.

[20] R. Schwitter, "A layered controlled natural language for knowledge representation," in *Machine Translation, Controlled Languages and Specialised Languages: Special Issue of Linguisticae Investigationes*, 2005, pp. 85–106.

A Human Action Descriptor based on Motion Coordination

Pietro Falco[1], Matteo Saveriano[1], Eka Gibran Hasany[2], Nicholas H. Kirk[1] and Dongheui Lee[1]

***Abstract*— In this paper, we present a descriptor for human whole-body actions based on motion coordination. We exploit the principle, well-known in neuromechanics, that humans move their joints in a coordinated fashion. Our coordination-based descriptor (CODE) is computed by two main steps. The first step is to identify the most informative joints which characterize the motion. The second step enriches the descriptor considering minimum and maximum joint velocities and the correlations between the most informative joints. In order to compute the distances between action descriptors, we propose a novel correlation-based similarity measure. The performance of CODE is tested on two public datasets, namely HDM05 and Berkeley MHAD, and compared with state-of-the-art approaches, showing promising recognition results.**

I. Introduction

In the last two decades, encoding and classifying human actions has been a key topic in computer vision and human movement science. Recently, motion interpretation has become a topic of great interest also within the robotic community. One of the challenges in modern robotics is to bring robots out of the structured industrial environments and let them work in close cooperation with humans. Robots will execute tasks in environments dwelled by humans and in direct contact with them. In order for robots to successfully interact with human beings, a necessary step is representing and classifying actions performed by humans.

In robotic applications, motion descriptors need to fulfill specific requirements of computational complexity and scalability in addition to accuracy. Modern autonomous robots have complex software architectures and very demanding planning and control algorithms. In order to make these systems usable in real world scenarios, it is essential to keep as low as possible the computational complexity, both for sake of time and energy consumption. Scalability is also an important issue, since in robotic applications the total number of actions and the duration of each action cannot be accurately predicted.

In order to take a step in matching these requirements, we propose a COordination-based action DEscriptor (CODE). CODE is characterized by a low time and space complexity, and achieves good scalability and classification accuracy. The concept of the proposed approach is shown in Fig.

This work has been supported by the Marie Sklodowska-Curie Indivisual Fellowship LEACON, EU project 659265, and by the Technical University of Munich, International Graduate School of Science and Engineering.

[1] Chair of Automatic Control Engineering, Technical University of Munich, Munich, Germany {pietro.falco, matteo.saveriano, nicholas.kirk, dhlee}@tum.de.

[2] Department of Informatics, Technical University of Munich, Munich, Germany eka.hasany@tum.de.

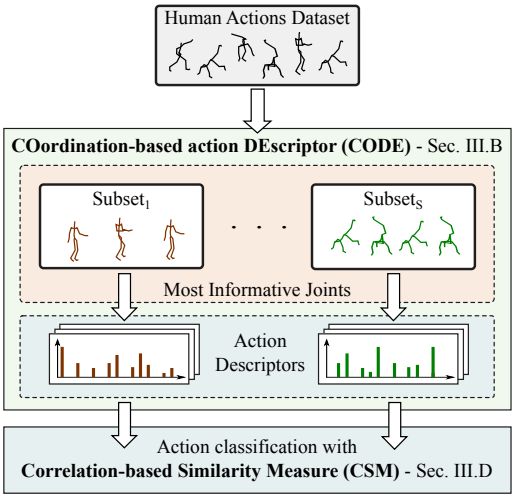


Fig. 1. Overview of the proposed approach for action representation and recognition. Intuitively, we can say that selecting the most informative joints splits the dataset into several action subsets. The actions within each subset have similar most informative joints. Neuromechanically-sound features are then added to make action descriptors more distinctive. Finally, action classification is performed using the proposed CSM metric.

1. CODE leverages the property of human motion, well-known in neuromechanics, that humans move their joints in a coordinated fashion [1]–[4] and that the various degrees of freedom present couplings and dependencies [2], [3]. One of the main contributions of this work is to exploit such correlation among joints to increase the performance of action recognition in terms of accuracy and scalability.

In CODE, therefore, information about correlation is a key tool to characterize motion. In order to reduce the computational complexity, CODE analyzes the correlation properties of a subset of joints, called most informative joints [5]. Roughly speaking, the most informative joints are the joints which mostly contribute to the execution of a certain action. CODE selects the most informative joints on the basis of the signal variances. In the literature concerning motion analysis, this assumption has been proven to be valid [5] for classification applications. To increase the discriminativeness of each action, we enrich the descriptor with information about motion coordination (correlation between joint pairs), and information about velocities to discriminate the directionality of motion. Moreover, we propose a novel similarity measure, called Correlation-based Similarity Measure (CSM), which performs better than the classical Euclidean and Manhattan distances with a reduced number of informative joints.

The rest of the paper is organized as follows. Section II presents the related work. Section III describes the proposed

action descriptor and similarity measure. Experiments on two human action datasets, namely Hochschule Der Medien 2005 (HDM05) [6] and Berkeley Multimodal Human Action Database (MHAD) [7], and a comparison with state-of-the-art approaches are shown in Sec. IV. Section V states conclusions and proposes future extensions.

II. Related Work

In the literature, there are diverse works on motion recognition, which are based on different types of input data. Two common representations of human motion are based on normalized joint positions and on joint angles. In Cartesian-based representations, motion is described with the positions of the joints in the 3D space expressed in a reference frame fixed to the human torso. As a consequence, a precise skeletal model is required for this representation. Representations based on joint angles, instead, are natively independent from the used reference frame [5]. Joint angles can be computed by inverse kinematics of a skeleton model or measured by wearable sensors such as inertial measurement units [8], [9]. This representation is potentially more interesting in robotics, since it does not implicitly assume the knowledge of the skeletal model and it does not require a normalization step. CODE is designed for angle-based representations, since the neuromechanical properties of human motion coordination have been proven for joint angles [1], [2].

Methods based on Joint Cartesian Positions: Cartesian trajectories are strongly affected by the choice of the reference frame and the link lengths, which reduces the discriminative power of Cartesian descriptors [10]–[12]. To alleviate this problem, a normalization procedure is performed [10], which expresses the joint positions in a frame fixed to the torso and normalizes the length of the bones. The method is defined skeleton-based (or model-based) because it requires the knowledge of the skeletal model of the performer to obtain a user-independent normalized representation. Using this skeleton-based representation, in [10] a deep neural network is proposed to classify motion capture sequences. In [13], a hierarchical recurrent neural network is proposed for action classification. A template-based approach to recognize actions [14] uses a small set of a-priori known actions called templates. To align observed actions with the templates, the dynamic time warping [15] is adopted. In [16], a local skeleton descriptor is proposed that encodes the relative position of joint quadruples. The input data are joint Cartesian coordinates. The approach in [17] exploits learned models to represent each action and to capture the intra-class variance. The method shows promising results in dealing with data from depth cameras. The work in [18] describes a representation based on pairwise joint-to-joint distances in the skeletal model and principal component analysis is used to reduce the dimensionality.

Methods based on Joint Angles: In [19], an online segmentation and recognition of manipulation task, based on singular value decomposition, is proposed. An unsupervised approach that exploits hidden Markov models to segment and recognize actions is presented in [20]. The work presented in [21] leverages the properties of human motion in frequency domain to derive a compact action descriptor. Linear Dynamical Systems (LDS) are used in [22] to recognize human gaits, and the methodology can be applied also to recognition of whole-body actions. In [5], the authors propose three descriptors ranking the most informative joints involved in an action, i.e. the joints which have highest variance during the motion. The descriptors are called Sequence of the Most Informative Joints (SMIJ), Histograms of Most Informative Joints (HMIJ) and Histogram of Motion Words (HMW), respectively. This approach is particularly significant for our work, since it proposes descriptors effective in discriminate actions but also simple and computationally efficient. This philosophy is also used in CODE as well as the concept of choosing the most informative joints based on the variance. There are two main differences between SMIJ [5] and CODE. First, CODE computes the variance of the overall motion trajectory (global descriptor) and has a constant size, while SMIJ requires to split each action into several segments (local descriptor). Second, we explicitly take into account motion coordination and propose a novel Correlation-based Similarity Measure (CSM) to compute the similarity between action descriptors. Recognition performance of LDS, HMW, SMIJ, HMIJ and CODE are compared in Sec. IV-D.

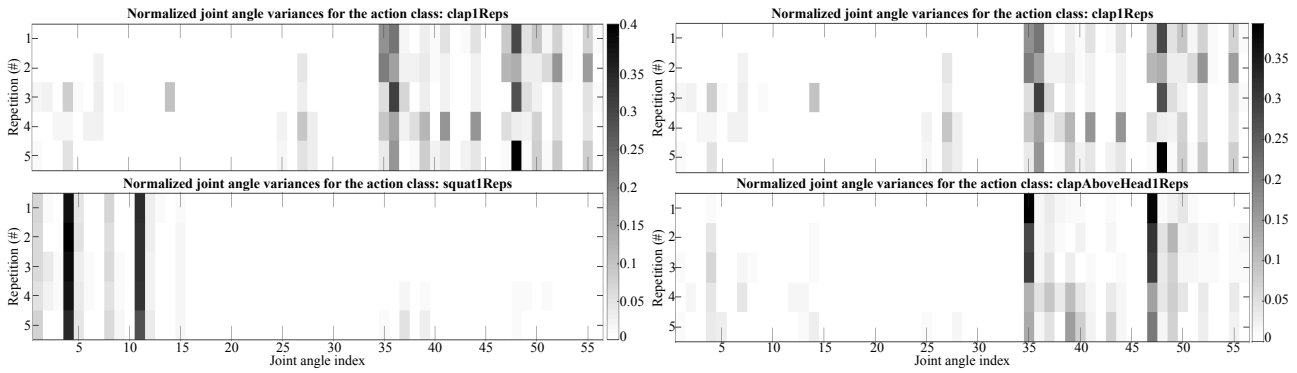
Aforementioned angle-based representations present two important open points. First, they are tested only on a limited set of classes (10-15 classes) and, therefore the scalability is not investigated. Second, the complexity analysis is usually neglected, even though it is an important theoretical foundation for real applicability. CODE, on the other hand, offers a good balance between accuracy, scalability, and computational complexity. CODE performs well not only on a typical datasets of 10-15 classes, but also on the whole HDM05 dataset, constituted by 80 classes and 2337 actions.

III. Proposed Approach

This section discusses three problems related to action classification: *i)* which raw data from tracking systems are better suited for action representation, *ii)* which features can be extracted from sensory data to reduce the dimensionality and increase the discriminativeness, and *iii)* how the similarity between actions can be measured.

A. Whole-body action representation

Modern motion tracking systems adopt a kinematic model of the human body, the so-called skeletal model, consisting of a certain number of links connected by joints. The raw information available from the tracking system is a time series of skeletal poses sampled at different time instants. A possible way to represent whole-body actions is to collect a set of 3D joint positions sampled at different times, i.e. a set of Cartesian trajectories. As discussed in Sec. II, Cartesian trajectories depend on the reference frame in which the motion is expressed and on the length of human limbs. On the other hand, joint angles between two connected links in the skeletal model are naturally invariant to roto-translations and scaling factors. Hence, in this work, we represent an action



(a) (b)

Fig. 2. Joint angle variances as a function of joint index and repetition number. (a) *clap1Reps* and *squat1Reps* have different sets of most informative joints. This kind of actions can be correctly classified considering only the relevant joints. (b) *clap1Reps* and *clapAboveHead1Reps* have similar sets of most informative joints. For this kind of actions misclassification may occur if only the most informative joints are considered as features.

as a set of joint angles trajectories, i.e. as the $J \times T$ matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_J]$, where $\boldsymbol{a}_j = \{a_j^t\}_{t=1}^T$ is the trajectory of the j-th joint angle, J is the number of joints and T is the number of time frames. One possibility is to directly use the raw time series $\boldsymbol{A}$ for action classification. Alternatively, as in this work, one can extract from $\boldsymbol{A}$ a feature vector (action descriptor) whose objective is to reduce the size of the input data and increase their discriminativeness.

B. Coordination-based action descriptor

The proposed action descriptor is based on two assumptions. The first assumption is that, while each subject can perform the same action in different manners generating different joint trajectories, all the subjects tend to activate the same set of joints [5]. For example, in a clapping action the arm joints are the most informative, while the rest are practically unused. The second assumption is that humans move the joints in a coordinated fashion [1], and, therefore, motion coordination is discriminative for motion recognition.

Building upon these assumptions, we define the CODE action descriptor $\mathcal{A}$ as the 5-tuple

$$\mathcal{A} \triangleq (I_m, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{v}}_{max}, \hat{\boldsymbol{v}}_{min}, \boldsymbol{c}) \tag{1}$$

where I_m contains the indexes of the J_m most informative joints (MIJ), $\hat{\boldsymbol{\sigma}} \in \mathbb{R}^{J_m}$, $\hat{\boldsymbol{v}}_{max} \in \mathbb{R}^{J_m}$ and $\hat{\boldsymbol{v}}_{min} \in \mathbb{R}^{J_m}$ are respectively the normalized variances, maximum and minimum velocities of the MIJ. The vector $\boldsymbol{c}$ is the correlation between each pair of MIJ and has $J_m(J_m-1)/2$ components. In more detail, the vector $\boldsymbol{c}$ is obtained by concatenating the correlation coefficients c_{ij}, where (i, j) is a couple of most informative joints of an action A. If an action has J_m most informative joints, we will have $J_m(J_m-1)/2$ pairwise combination. With the symbols $\mathcal{A}$ we denote a finite ordered list of elements (a tuple). Each element of this tuple is a vector. For implementation purposes, the elements of the 5-tuple $\mathcal{A}$ are stacked into an array of $N_C = J_m(J_m + 7)/2$. components. Hence, the number of MIJ J_m determines the size of the descriptor and it has to be chosen in order to guarantee a good compromise between dimensionality (computation time) and recognition performance. Details about the action descriptor in (1) are provided in the rest of this Section.

1) Selecting the most informative joints: During the execution of an action, not all the joints contribute in the same manner. Hence, a possible way to represent a motion is to find which joints contribute the most to the whole motion, i.e. which are the most informative joints (MIJ). The variance $\sigma_j, j = 1, \ldots, J$ of each joint angle trajectory is used to identify the $J_m \leq J$ most informative joints, considering that the higher the variance, the higher the contribution of that joint to the whole-body motion [5].

For a given action $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_J]$, the variance is computed for all the J columns of $\boldsymbol{A}$, obtaining the vector $\boldsymbol{\sigma}^a = [\sigma_1^a, \ldots, \sigma_J^a]^T$. The elements of $\boldsymbol{\sigma}^a$ are sorted as

$$\begin{aligned} &(\boldsymbol{\sigma}^s, I^s) = \text{sort}\,(\boldsymbol{\sigma}^a)\,, \\ &\boldsymbol{\sigma}^s = [\sigma_1^s, \sigma_2^s, \ldots, \sigma_{J_m}^s, \ldots, \sigma_J^s]^T, \\ &I^s = \{i_1^s, i_2^s, \ldots, i_{J_m}^s, \ldots, i_J^s\} \end{aligned} \tag{2}$$

where the function $\text{sort}\,(\boldsymbol{u})$ sorts the elements of $\boldsymbol{u}$ in descending order and returns the sorted indexes I^s. The vector of normalized variances $\hat{\boldsymbol{\sigma}}$ of the J_m MIJ is computed as

$$\begin{aligned} &I_m = \{i_1^s, i_2^s, \ldots, i_{J_m}^s\}, \\ &\boldsymbol{\sigma} = [\sigma_1^s, \sigma_2^s, \ldots, \sigma_{J_m}^s]^T, \\ &\hat{\boldsymbol{\sigma}} = \frac{\boldsymbol{\sigma}}{\sum_{j=1}^{J_m} \sigma_j^s} = [\hat{\sigma}_1, \hat{\sigma}_2, \ldots, \hat{\sigma}_{J_m}]^T \end{aligned} \tag{3}$$

The last expression in (3) guarantees that $\sum_{j=1}^{J_m} \hat{\sigma}_j = 1$. It is worth noticing that taking the variance of the MIJ $\hat{\boldsymbol{\sigma}}$ as action descriptor significantly reduces the amount of data. Indeed, as discussed in Sec. III-A, raw sensory data are $T \times J$ matrices, where T is usually bigger than J, while $\hat{\boldsymbol{\sigma}}$ is a vector with $J_m \leq J$ components. In this work, we set $J_m = 20$, as motivated in Sec. IV-B.

The colormaps in Fig. 2 represent the normalized joint angle variances $\hat{\boldsymbol{\sigma}}^a = \boldsymbol{\sigma}^a / \sum_j^J \sigma_j^a$ as a function of the joint angle index. Three action classes are considered from

the HDM05 database: *clap1Reps*, *clapAboveHead1Reps* and *squat1Reps*. Each action is repeated 5 times, and each repetition is associated to a repetition number. Let us firstly focus on a single action class, e.g. *squat1Reps* in Fig. 2(a). Each row of the colormap represents a repetition of *squat1Reps*. We can see that only a small subset of joints have not negligible variance and all the repetition have a common set of informative joints. Moreover, in Fig. 2, the class *clap1Reps* is compared, in terms of joint angle variances, with *squat1Reps* in Fig. 2(a) and with *clapAboveHead1Reps* in Fig. 2(b). Looking at the figure, it is evident how actions that use different MIJ, such as *squat1Reps* and *clap1Reps*, present a different joint variance pattern (see Fig. 2(a)). On the other hand, classes like *clapAboveHead1Reps* and *clap1Reps*, which have similar MIJ, present a similar variance pattern, as shown in Fig. 2(b).

MIJ can easily discriminate actions executed with different joints. Nevertheless, when dealing with large datasets, different classes with similar MIJ can become very common. To increase the discriminativeness, we enrich our descriptor with velocities and pairwise correlations between the MIJ.

2) Maximum and minimum velocity of the MIJ: The variance captures information on joint angular motion without considering the direction of the motion. Distinguishing between positive and negative joint rotations increases the informativeness of the descriptor and improves the recognition performance. The normalized maximum and minimum MIJ velocities

$$\hat{\boldsymbol{v}}_{max} = \frac{\boldsymbol{v}_{max}}{\sum_{j=1}^{J_m} |v_{max,j}|}, \quad \hat{\boldsymbol{v}}_{min} = \frac{\boldsymbol{v}_{min}}{\sum_{j=1}^{J_m} |v_{min,j}|} \tag{4}$$

are also considered in our descriptor. By construction, $\hat{\boldsymbol{v}}_{max}$ and $\hat{\boldsymbol{v}}_{min}$ are vectors with J_m components.

3) Pairwise correlation of the MIJ: Neuromechanical evidences show a certain degree of correlation between the most informative joints (or a subset of MIJ) [1], [2]. To exploit such a correlation, we enrich the descriptor with the vector $\boldsymbol{c}$ of pairwise correlations of the J_m most informative joints. In particular, given a MIJ trajectory $\boldsymbol{A}_m = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{J_m}] \in \mathbb{R}^{T \times J_m}$, one can compute the pairwise correlation matrix

$$\boldsymbol{C} = \begin{bmatrix} 1 & c_{1,2} & \cdots & c_{1,J_m} \\ c_{2,1} & 1 & \cdots & c_{2,J_m} \\ \vdots & \vdots & \ddots & \vdots \\ c_{J_m,1} & c_{J_m,2} & \cdots & 1 \end{bmatrix} \tag{5}$$

where the element $-1 \leq c_{ij} \leq 1$ represents the linear correlation between the joint i and j and it is computed as

$$c_{ij} = \frac{\sum_{t=1}^{T}(a_i^t - \bar{a}_i)(a_j^t - \bar{a}_j)}{\sqrt{\sigma_i^s}\sqrt{\sigma_j^s}} = \frac{\text{cov}(\boldsymbol{a}_i, \boldsymbol{a}_j)}{\sqrt{\sigma_i^s}\sqrt{\sigma_j^s}} \tag{6}$$

The quantities $\bar{a}_i$ and $\bar{a}_j$ in (6) are the mean values of $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$ respectively, while the variances σ_i^s and σ_j^s are defined as in (3). The numerator of (6) represents the covariance between $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$. By construction, the correlation matrix $\boldsymbol{C}$ in (5) is symmetric with unitary diagonal elements. The $J_m(J_m-1)/2$ different entries in $\boldsymbol{C}$ are stacked into the correlation vector $\boldsymbol{c}$ and used to augment our descriptor. The procedure to compute CODE is summarized in Algorithm 1.

Algorithm 1 CODE Descriptor

input: Action matrix $\boldsymbol{A}$, MIJ number J_m

1: Compute normalized variance and MIJ indexes
$\boldsymbol{\sigma}^a = \text{variance}\,(\boldsymbol{A})$
$(\boldsymbol{\sigma}^s, I^s) = \text{sort}\,(\boldsymbol{\sigma}^a)$
$\boldsymbol{\sigma} = [\sigma_1^s, \sigma_2^s, \ldots, \sigma_{J_m}^s]^T$
$I_m = \{i_1^s, i_2^s, \ldots, i_{J_m}^s\}$
$\hat{\boldsymbol{\sigma}} = \boldsymbol{\sigma} / \sum_{j=1}^{J_m} \sigma_j^s$
2: Compute normalized velocities
$\hat{\boldsymbol{v}}_{max} = \boldsymbol{v}_{max} / \sum_{j=1}^{J_m} v_{max,j}$
$\hat{\boldsymbol{v}}_{min} = \boldsymbol{v}_{min} / \sum_{j=1}^{J_m} v_{min,j}$
3: Compute correlation vector
$\boldsymbol{C} = \{c_{ij}\}_{i=1,j=1}^{i=J_m,j=J_m}$, where $c_{ij} = \text{cov}(\boldsymbol{a}_i, \boldsymbol{a}_j)/(\sqrt{\sigma_i^s}\sqrt{\sigma_j^s})$
stack the upper (or lower) triangular part of $\boldsymbol{C}$ into the vector $\boldsymbol{c}$
4: **return** $[I_m, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{v}}_{max}, \hat{\boldsymbol{v}}_{min}, \boldsymbol{c}]$

C. Analysis of Space and Time Complexity

We report in Table I the (computational) time and space complexity of the CODE descriptor as a function of the number of most informative joints J_m and the number of action time frames T. As described previously in this section, CODE has $J_m(J_m+7)/2$ components. Hence, using the big O notation [23], its space complexity is $\mathcal{O}(J_m^2)$. The space complexity is $\mathcal{O}(1)$, since the size of CODE is independent from the number of time frames T. Regarding the time complexity as a function of J_m, the most time-complex operation in Algorithm 1 is step 3, i.e., computation of the correlation vector. The computation of the correlation coefficient is performed as in (6) for each pair of MIJ. Since there are $J_m(J_m-1)/2$ combinations of MIJ pairs, the time complexity as a function of J_m is $\mathcal{O}(J_m^2)$. The time complexity as a function of the number of time frames is $\mathcal{O}(T)$, since the computation of variances in (3), the computation of normalized velocities in (4), and the computation of the correlation vector in (6) have $\mathcal{O}(T)$ time complexity. Overall, CODE has $\mathcal{O}(J_m^2 T)$ time complexity and $\mathcal{O}(J_m^2)$ space complexity.

	Time Complexity	**Space Complexity**
MIJ number (J_m)	$\mathcal{O}(J_m^2)$	$\mathcal{O}(J_m^2)$
Frames (T)	$\mathcal{O}(T)$	$\mathcal{O}(1)$
Overall (J_m, T)	$\mathcal{O}(J_m^2 T)$	$\mathcal{O}(J_m^2)$

TABLE I
TIME AND SPACE COMPLEXITY OF CODE AS A FUNCTION OF THE NUMBER OF MIJ J_m AND THE NUMBER OF FRAMES T.

D. Correlation-based similarity measure

As described in Sec. III-B, CODE represents an action with a vector of dimension N_C. To measure the similarity

among actions, we propose a novel similarity measure called Correlation-based Similarity Measure (CSM).

Consider the two action descriptors $\mathcal{A}^a$ and $\mathcal{A}^b$ where $\mathcal{A}^u = (I^u_m, \hat{\boldsymbol{\sigma}}^u, \hat{\boldsymbol{v}}^u_{max}, \hat{\boldsymbol{v}}^u_{min}, \boldsymbol{c}^u)$, $u = a, b$. Let us define the set $\mathcal{S} = \{(i,j) \in I^a_m \cap I^b_m | i \neq j\}$. In practice, $\mathcal{S}$ contains the pairs of MIJ that are common to $\mathcal{A}^a$ and $\mathcal{A}^b$. The CSM between two action descriptors $\mathcal{A}^a$ and $\mathcal{A}^b$ is defined as

$$\begin{aligned} CSM(\mathcal{A}^a, \mathcal{A}^b) = \sum_{i,j \in \mathcal{S}} w_{ij} [& (\hat{\sigma}^a_i + \hat{\sigma}^a_j + \hat{\sigma}^b_i + \hat{\sigma}^b_j) + \\ & + (\hat{v}^a_{max,i} + \hat{v}^a_{max,j} + \hat{v}^b_{max,i} + \hat{v}^b_{max,j}) \\ & + (\hat{v}^a_{min,i} + \hat{v}^a_{min,j} + \hat{v}^b_{min,i} + \hat{v}^b_{min,j})] \end{aligned} \tag{7}$$

where the weight $w_{ij} = 1 - 0.5|c^a_{ij} - c^b_{ij}|$ is maximum ($w_{ij} = 1$) when the action a and b have the same correlation between the common most informative joints i and j. The weight w_{ij} is minimum ($w_{ij} = 0$) if the common MIJ i and j are perfectly correlated in action a ($c^a_{ij} = 1$) and anti-correlated in action b ($c^b_{ij} = -1$), or viceversa ($c^a_{ij} = -1$ and $c^b_{ij} = 1$). The correlation-based similarity measure in (7) is a summation of variances and velocities of common MIJ weighted by the differences in pairwise correlations between the two actions. Hence, two actions which use the same MIJ, but are characterized by a different correlation pattern, will have a low CSM score. High values of CMS indicate a high similarity between two actions. CSM is zero if two actions have no common MIJ or if all the MIJ are anti-correlated. Moreover, the joints that present a higher variance, maximum and minimum velocities give more contribution to the evaluation of similarity CSM than joint with low variance, and velocities. Figure 3 shows the value of the weight w_{ij} for two actions a and b as a function of the difference in correlation between two common most informative joints i and j.

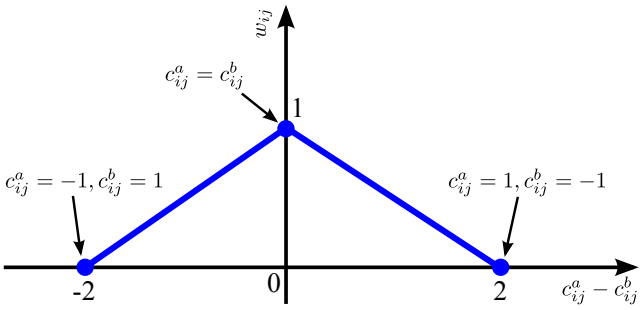


Fig. 3. Value of the weight w_{ij} as a function of $c^a_{ij} - c^b_{ij}$.

IV. Experimental Results

In order to prove the effectiveness of our approach, we perform three types of experiments on the public motion datasets HDM05 [6] and MHAD [7]. In the first type of experiments, we evaluate the accuracy on the whole HDM05 dataset as a function of the number of most informative joints with different features and different similarity measures. In the second set of experiments, we evaluate accuracy, precision and recall of CODE. The third class of experiments consists in a comparison with other descriptors in the literature. In order to reduce high-frequency noise, we apply a butterworth filter with cut-off frequency of 10 Hz.

A. Dataset description

We use three different datasets for our experiments: (i) HDM05, (ii) Reduced HDM05 and (iii) MHAD. The main characteristics of each dataset are summarized in Table II.

The HDM05 dataset contains 2337 actions split into 130 classes, and the actions are performed by 5 subjects. We consider 80 classes obtained by merging the motion recordings that contain multiple executions of the same action. For example, clap one repetition and clap five repetitions have been considered to be in the same class.

The Reduced HDM05 (R-HDM05) dataset is a subset of HDM05 composed by 401 action sequences split into the 16 classes: "emphdepositFloorR (1), *elbowToKnee1RepsLelbowStart* (2), *grabHighR* (3), *hopBothLegs1hops* (4), *jogOnPlaceStartAir2StepsLStart* (5), *jumpDown* (6), *jumpingJack1Reps* (7), *kickLFront1Reps* (8), *lieDownFloor* (9), *rotateArmsBothBackward1Reps* (10), *sitDownChair* (11), *sneak2StepsLStart* (12), *squat1Reps* (13), *standUpKneelToStand* (14), *throwBasketball* (15), *throwFarR* (16)". The numbers in brackets are the class labels used in Fig. 6. These are the action classes chosen in [5], which we adopt to perform comparisons.

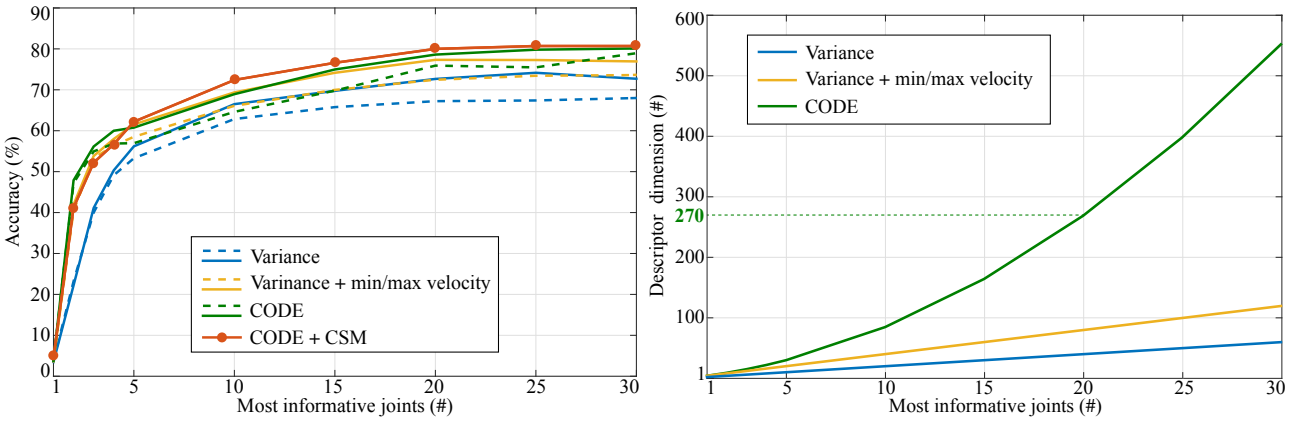
MHAD is constituted by 11 classes:"*jumping* (1), *jumping jacks* (2), *bending* (3), *punching* (4), *waving two hands* (5), *waving one hand* (6), *clapping* (7), *throwing* (8), *sit down* (9), *stand up* (10), *sit down/stand up* (11)". The numbers in brackets are class labels used in Fig. 7. Each action is performed by 12 subjects 5 times, yielding a total of 659 actions (1 erroneous action was removed from the database).

B. Number of most informative joints

The goal of this experiment is two-fold. First, it shows the contribution of the different CODE components in Sec. III-B. Second, it investigates how to choose an efficient number of most informative joints. To guarantee a statistical relevance, we tried CODE on a large set of actions and classes, i.e., the 80 classes and 2337 actions of HDM05. The accuracy of CODE, evaluated as a function of the Most Informative Joints (MIJ) number J_m, is shown in Fig. 4(a). The accuracy is computed as the ratio between the number of total test inputs correctly classified and the number of test inputs. In the figure, CODE with the proposed CSM is compared with descriptors based (i) only on variance of MIJ, (ii) on variance

Dataset	Subjects (#)	Classes (#)	Actions (#)	Frame Rate (Hz)
HDM05	5	80	2337	120
R-HDM05	5	16	401	120
MHAD	12	11	659	480

TABLE II
DATASETS CHARACTERISTICS.



(a) Accuracy as a function of MIJ number. Dashed lines are obtained with Euclidean distance, solid lines with Manhattan distance.

(b) Descriptors dimension as a function of MIJ number.

Fig. 4. Results on the HDM05 dataset (2337 actions and 80 classes). (a) Recognition results for different values of J_m and different features vectors. (b) Motion descriptors that consider only variance or variance and velocity as features grow linearly with J_m, while CODE grows quadratically. CODE with CSM offers a good compromise between recognition rate (80.0%) and descriptor dimension (270 components with $J_m = 20$).

and joint angular velocities of MIJ, (iii) on variance, velocity, and correlation of MIJ. The results show that all CODE features contribute to improve the recognition rate.

The continuous lines in Fig. 4(a) denote the use of Manhattan distance, while the dashed lines denote Euclidean distance to evaluate the similarity between actions. In case of CODE+CSM, we use our proposed metrics to evaluate the similarity. We can see that, in general, Manhattan distance performs better than Euclidean, and CSM performs better than Manhattan distance for $J_m \geq 5$. An advantage of the proposed Correlation-based Similarity Measure is that CODE+CSM performs better with less MIJ with respect to Euclidean and Manhattan distances. For example, with $J_m = 20$, CODE+CSM achieves 80.0% of accuracy, while CODE+Manhattan achieves 78.6% of accuracy. When increasing the number of MIJ ($J_m \geq 20$), the difference between the metrics becomes smaller. For example, with $J_m = 30$, CODE+CSM achieves 80.7% of accuracy, while CODE+Manhattan achieves 80.1% of accuracy. We can conclude that CSM achieves better performance than Euclidean and Manhattan distances with a reduced number of MIJ. Figure 4(b) shows the dimension of CODE as a function of the number of most informative joints. The dimension of CODE increases quadratically with J_m. This is an expected result considering the spatial complexity analysis in Sec. III-C. Using only variance and variance+velocities, the size of the descriptor increases linearly. The price paid for a more precise characterization of the motion is an increase in the descriptor dimensionality. Considering the accuracy in Fig. 4(a) (80.0% with $J_m = 20$ and 80.7% with $J_m = 30$) and the descriptor size in Fig. 4(b) (270 components with $J_m = 20$ and 555 components with $J_m = 30$), we can conclude that CODE+CSM with $J_m = 20$ offers a good compromise between recognition rate and size of the descriptor.

C. Performance Evaluation

Using 10-fold cross-validation, accuracy, precision, and recall of CODE have been evaluated on three datasets: HMD05, R-HMD05, and MHAD. Precision is obtained as the ratio between true positives and the sum of true positives and false positives. Recall is obtained as the ratio between true positives and the sum of true positives and false negatives. Also, we report the time to compute CODE for all the actions of each dataset. The computer used for the evaluation has an Intel® Core™ $i7-4790\,\mathrm{K}$ - 4 Cores CPU, and $16\,\mathrm{GB}$ of memory. CODE is implemented in Matlab® 2014b. The results, summarized in Table III, are obtained using CODE with CSM, $J_m = 20$ and 1-NN classification. The average accuracy of CODE on HDM05 is 80.0%, precision is 73.7% and recall is 73.0%. The time to compute the CODE for all the actions of HDM05 is $3.84\,\mathrm{s}$ with our unoptimized Matlab implementation. For the R-HDM05 dataset, we achieve the average accuracy of 96.0%, the average precision of 94.5%, and the average recall of 95.6%. The time to compute the descriptor for all action of R-HDM05 is $0.64\,\mathrm{s}$. In the experiments on the MHAD dataset accuracy, precision, and recall are 96.4%, 96.7% and 96.8%, respectively, while the time to compute CODE for all the actions is $9.54\,\mathrm{s}$. In Fig. 5, the robustness of CODE in presence of Additive Gaussian White Noise (AGWN) is

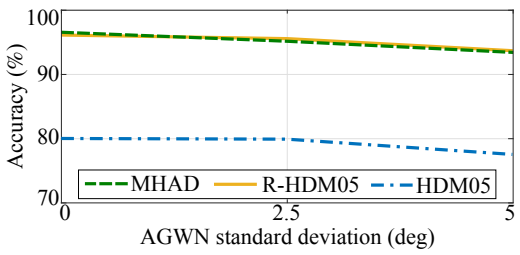


Fig. 5. Accuracy of CODE for different values of the AGWN standard deviation, evaluated on HDM05, R-HDM05, and MHAD.

Dataset	Accuracy (%) (mean±std)	Precision (%) (mean±std)	Recall (%) (mean±std)	Time (s) (mean±std)
MHAD	96.4 ± 2.9	96.7 ± 3.3	96.8 ± 2.5	9.54 ± 0.61
R-HDM05	96.0 ± 2.7	94.5 ± 3.5	95.6 ± 3.8	0.64 ± 0.02
HDM05	80.0 ± 2.9	73.7 ± 2.6	73.0 ± 2.7	3.84 ± 0.1

TABLE III

CROSS-VALIDATED (10-FOLD) RESULTS WITH CODE+CSM.

Descriptor	Classification	Accuracy (%)
CODE + CSM	1-NN	**98.4**
SMIJ [5]	1-NN	91.5
HMIJ [5]	1-NN	73.5
HMW [5]	1-NN	77.4
LDSP [5], [22]	1-NN	67.8

TABLE IV

CLASSIFICATION RESULTS FOR THE R-HDM05 DATASET.

evaluated. We corrupted the joint angle signals with AGWN of standard deviation in the range $[0,\ 5]$ deg. For R-HDM05, with a standard deviation of 5 deg the accuracy is 93.8%, for MHAD is 93.3%, while for HMD05 the accuracy is 77.5%. Roughly, we loose about 3% accuracy corrupting the signals with additional AGWN of 5 deg standard deviation.

D. Comparison with angle-based approaches

We compare the recognition performance of CODE with the state-of-the-art descriptors in [5], [16], [22]. The comparison is carried out on both the R-HDM05 and the MHAD datasets. As in the previous experiments, we use CSM to measure the similarity between the CODE descriptors of different actions and $J_m = 20$ most informative joints.

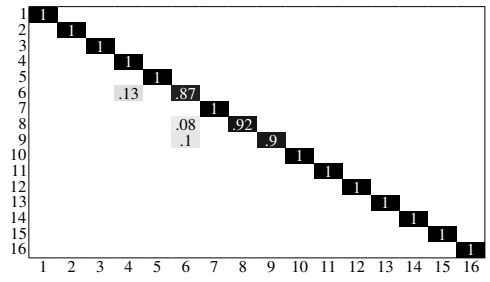


Fig. 6. Confusion matrix for 1-NN classification of the R-HDM05 dataset.

R-HDM05: For a fair comparison, we adopt the same 16 classes (see Sec. IV-A) and the same cross-subject validation protocol used in [5]. In particular, we consider 3 subjects (219 action sequences) for training and the remaining 2 subjects (182 action sequences) for testing. Cross-subject validation is particularly interesting to demonstrate the generalization capabilities of CODE across different users. Additionally, we compare CODE with Histograms of Most Informative Joints (HMIJ) [5], Histogram-of-Motion Words (HMW) [5], and Linear Dynamical System Parameter (LDSP) [22]. The results of this comparison are shown in Table IV. We can see that the best results are achieved by CODE, with an accuracy of 98.4%. The confusion matrix relative to this case study in presented in Fig. 6. The actions that do not achieve 100.0% accuracy are *jumpDown* (6), *kickLFront1Reps* (8), *lieDownFloor* (9). The action *jumpDown* has 87.0% accuracy and is confused with *hopBothLegs1hops* (4) in 13.0% of cases. The accuracy for *kickLFront1Reps* (8) is 92.0% and it is confused with *jumpDown* (6). *lieDownFloor* (9), which presents an accuracy of 90.0%, is confused with *jumpDown* (6) in the 10.0% of cases.

MHAD: The comparison between CODE, SMIJ, HMIJ, and LDSP on the classes of the MHAD database is reported in Table V. In this case, CODE achieves 98.5% accuracy and the second best is SMIJ that achieves 94.5%. In this experiment, 7 subjects are chosen for training (384 action sequences) and 5 (275 action sequences) for testing, according to the cross-subject validation protocol adopted in [5]. The confusion matrix is shown in Fig. 7. We can see that the accuracy of CODE is 100.0% for the majority of the classes, except for three classes: *jumping* (1), *sit down* (10), *sit down and stand up* (11). The accuracy is 96.0% for the action *jumping* (1), which has been confused in 4.0% of cases with the action *jumping jacks* (2). Moreover, the action *sit down* (10) presents a recognition rate of 92.0%, since it is confused in 4.0% of cases with *stand up* (9), and in 4.0% of cases with *sit down and stand up* (11). The action *sit down and stand up* achieves 96.0% accuracy and it is confused with *sit down* in 4.0% of cases.

Descriptor	Classification	Accuracy (%)
CODE + CSM	1-NN	**98.5**
SMIJ [5]	1-NN	94.5
HMIJ [5]	1-NN	80.3
HMW [5]	1-NN	77.7
LDSP [5], [22]	1-NN	84.9

TABLE V

CLASSIFICATION RESULTS FOR THE MHAD DATASET.

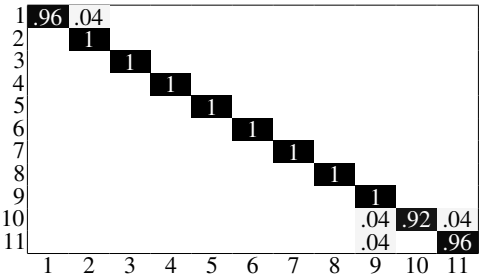


Fig. 7. Confusion matrix for 1-NN classification of the MHAD dataset.

E. Comparison with position-based approaches

In addition to the comparison with angle-based methods, we compare CODE also with approaches that use joint Cartesian positions. The two representations work with different input data, i.e. joint angles and 3D joint positions,

respectively. Since the recognition performance strongly depends on the type of input data, the comparisons in terms of accuracy are merely indicative. However, the scope of this section is to discuss basic differences between CODE and most successful position-based recognition approaches. First, we compared CODE with the skeleton quad descriptor presented in [16]. This approach obtains 93.89% on a subset (11 classes) of the R-HDM05 dataset. On the same subset, CODE achieves 100% accuracy. The second comparison is with the template-based approach (TBA) presented in [14]. It adopts DTW [15] to align the training trajectories with the test trajectories and has been tested with 9 classes [14] of HDM05 dataset, achieving 98.0% accuracy. On the same classes CODE achieves 98.3% accuracy. In terms of accuracy, the performance of TBA and CODE are similar on the tested classes. However, TBA has a $\mathcal{O}(T)$ spatial complexity (to store the entire joint position trajectories) and $\mathcal{O}(T^2)$ time complexity (to align training and test trajectories with DTW), while CODE has $\mathcal{O}(1)$ spatial complexity and $\mathcal{O}(T)$ time complexity (see Table I). The third comparison is with the skeleton-based approach (SKA) in [10]. It uses a deep neural network and a frame-by-frame classification to recognize motion capture sequences. The experiments are performed on 2337 actions of HDM05 split in 65 classes, achieving 95.6% accuracy. On the same action set CODE achieves 87.7% accuracy. In terms of accuracy SKA performs better than CODE. However, SKA uses a more complex descriptor with $33 \times T$ elements, where T is the number of time frames. The space complexity is therefore $\mathcal{O}(T)$, while CODE has a fixed size of 270×1 elements. Moreover, SKA adopts a classification algorithm based on deep learning, which requires a relatively long training time, while in this work we use a 1-NN classifier to keep the system simple and fast, according to the requirements typical of robotic systems.

V. Conclusions and Future Work

In this work, we presented CODE, a COordination-based action DEscriptor. CODE is based on the assumption, accepted in neuromechanics, that humans move in a coordinated fashion. CODE encodes the coordination properties of human motion by computing the pairwise correlations between the most informative joints. With experiments on two different datasets containing a large set of actions, we have shown that, including information about correlation and about joint velocities, the recognition performance improves significantly. The size of CODE is independent from the action duration and increases quadratically with the number of most informative joints. The comparisons showed that CODE outperforms several approaches for action recognition.

Future work will consist in evaluating CODE on representations based on Cartesian joint positions. Most renowned works in neuromechanics, in fact, discuss human motion correlation at a joint angle level. Therefore, the possibility to encode joint Cartesian positions with CODE-like descriptors requires further investigation. In order to segment streams of data before the classification, CODE can be combined with a state-of-the-art segmentation method such as [24]. A future work direction will consist in applying the basic concept of CODE also to the segmentation problem.

References

[1] N. A. Bernstein, *The coordination and regulation of movements*. Permagon Press, 1967.

[2] M. L. Latash, J. P. Scholz, and G. Schoner, "Toward a new theory of motor synergies," *Motor Control*, vol. 11, no. 3, pp. 276–308, 2007.

[3] S. Ambike and J. P. Schmiedeler, "The leading joint hypothesis for spatial reaching arm motions," *Experimental brain research*, vol. 224, no. 4, pp. 591–603, 2013.

[4] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *The journal of Neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[5] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 24–38, 2014.

[6] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber, *Documentation Mocap Database HDM05*, University of Bonn, 2007.

[7] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Berkeley MHAD: A comprehensive multimodal human action database," in *IEEE Workshop on Applications of Computer Vision*, 2013, pp. 53–60.

[8] P. Falco, G. De Maria, C. Natale, and S. Pirozzi, "Data fusion based on optical technology for observation of human manipulation," *Int. Journal of Optomechatronics*, vol. 6, no. 1, pp. 37–70, 2012.

[9] K. J. O'Donovan, R. Kamnik, D. T. OKeeffe, and G. M. Lyons, "An inertial and magnetic sensor based technique for joint angle measurement," *J Biomechs*, vol. 40, no. 12, pp. 2604–2611, 2007.

[10] K. Cho and X. Chen, "Classifying and visualizing motion capture sequences using deep neural networks," in *International Conference on Computer Vision Theory and Applications*, 2014, pp. 122–130.

[11] R. Soloperto, M. Saveriano, and D. Lee, "A bidirectional invariant representation of motion for gesture recognition and reproduction," in *Int. Conference on Robotics and Automation*, 2015, pp. 6146–6152.

[12] M. Saveriano and D. Lee, "Invariant representation for user independent motion recognition," in *International Symposium on Robot and Human Interactive Communication*, 2013, pp. 650–655.

[13] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1110–1118.

[14] D. Leightley, B. Li, J. S. McPhee, M. H. Yap, and J. Darby, "Exemplar-based human action recognition with template matching from a stream of motion capture," in *Int Conf on Image An Recog.*, 2014, pp. 12–20.

[15] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978.

[16] G. Evangelidis, G. Singh, and R. Horaud, "Skeletal quads: Human action recognition using joint quadruples," in *International Conference on Pattern Recognition*, 2014, pp. 4513–4518.

[17] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1290–1297.

[18] T. Le Naour, N. Courty, and S. Gibet, "Fast motion retrieval with the distance input space," in *Motion in Games*, 2012, pp. 362–365.

[19] A. Cavallo and P. Falco, "Online segmentation and classification of manipulation actions from the observation of kinetostatic data," *Trans. on Human-Machine Systems*, vol. 44, no. 2, pp. 256–269, 2014.

[20] D. Kulić, C. Ott, D. Lee, J. Ishikawa, and Y. Nakamura, "Incremental learning of full body motion primitives and their sequencing through human motion observation," *The International Journal of Robotics Research*, vol. 31, no. 3, pp. 330–345, 2012.

[21] D. Shah, P. Falco, M. Saveriano, and D. Lee, "Encoding human actions with a frequency domain approach," in *International Conference on Intelligent Robots and Systems*, 2016.

[22] A. Bissacco, A. Chiuso, Y. Ma, and S. Soatto, "Recognition of human gaits," in *Conf Comp Vision and Pattern Rec*, vol. 2, 2001, pp. 52–58.

[23] D. E. Knuth, "Big omicron and big omega and big theta," *ACM Sigact News*, vol. 8, no. 2, pp. 18–24, 1976.

[24] J. Barbič, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, and N. S. Pollard, "Segmenting motion capture data into distinct behaviors," in *Proceedings of Graphics Interface*, 2004, pp. 185–194.

Towards Learning Object Affordance Priors from Technical Texts

Nicholas H. Kirk[1]

***Abstract*— Everyday activities performed by artificial assistants can potentially be executed naïvely and dangerously given their lack of common sense knowledge. This paper presents conceptual work towards obtaining prior knowledge on the usual modality (passive or active) of any given entity, and their affordance estimates, by extracting high-confidence ability modality semantic relations (*X can Y* relationship) from non-figurative texts, by analyzing co-occurrence of grammatical instances of subjects and verbs, and verbs and objects. The discussion includes an outline of the concept, potential and limitations, and possible feature and learning framework adoption.**

I. Concept

In the domain of autonomous robot control, artificial assistants require to know what actions can be executed on a given set of objects. Such information, defined as *object affordances*, is usually obtained online by reinforcement or active learning during the execution of actions by processing percepts [1]. However, for safe human-robot interaction, we require the robot to have, from initialization, an understanding of *what actions an object can execute*, and *what actions an object can be subject to*. In this scope, we claim human-written technical texts can be an informative source to construct such initial world estimate. Such probability distribution over action-object relationships from natural language text can be performed thanks to the co-occurence understanding of verb-noun pairs: this analysis is known in computational linguistic literature as the use of distributional information of text to characterize lexical semantics, by considering statistical co-occurrence of neighbouring words [2]. However, the majority of current approaches make use of shallow syntactic features, which meaningfulness is debatable for semantic characterization [3]. We therefore make use of grammatical features, for partial semantic characterization of object affordances. While other semantic relationships employed in engineering are not easily prone to confident, automatic extraction and knowledge engineers have to recur to manual ontology insertion [4], the author's claim is that potentiality relationships can be robustly extracted from grammar relationships of Subject-Verb-Object (SVO) co-occurrences. The choice of the ability modality relationship calls for the assumption that the training corpus from which we derive data has to have reliable, non-figurative subject-verb-object co-occurrence tuples. More formally, co-occurrence of every noun $s_1 \in S$ with a verb $v_1 \in V$ entails the ability of s_1 to perform such action v_1 on the co-occurring object $o_1 \in O$. In simpler terms, we assume the instance "a robot builds a desk" implies "*a robot can build*" and "*a desk is buildable*", which the author does not consider a restrictive assumption that requires controlled authoring.

[1] Computer Science Department, Technische Universität München, Germany nicholas.kirk@tum.de

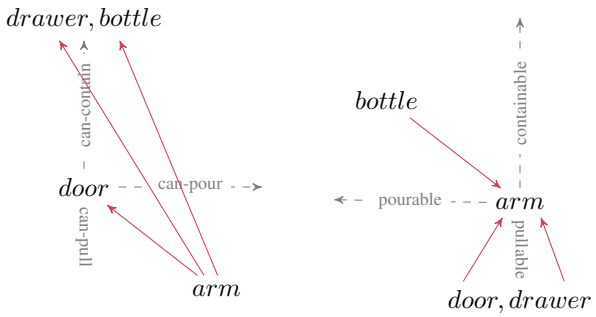


Fig. 1: Hypothetical representation of a dual active/passive 3-dimensional modality space (with predicates as dimensions) representing instances of kitchen scenario objects.

We therefore model our symbolic knowledge on potentiality as the joint probability distribution of all SVO occurrences in our training source, obtained via learning on typed dependency analysis output features of such source (see Equation 1).

$$Modality(W) = P\left(S \times V \times O\right) \tag{1}$$

$$S = \{\forall s \in N \mid grammar\_type(s, subject)\}$$
$$V = \{\forall v \in N \mid grammar\_type(v, verb)\}$$
$$O = \{\forall o \in N \mid grammar\_type(o, object)\}$$

From Equation 1 we derive two dual joint probability distributions, which encapsulate knowledge of active and passive noun roles (Eq. 2 and Eq. 3) and can induce two distinct vector spaces, representing passive and active role information (example in Figure 1).

$$Modality_{\text{active}}(W) = P(S \times V) \tag{2}$$
$$Modality_{\text{passive}}(W) = P(V \times O) \tag{3}$$

II. Implementation

In order to learn our distribution in Equation 1, a possible approach is to exploit Markov Logic Networks (MLN) [5] on a set of previously extracted Stanford typed dependencies [6]. The latter are a labeled, directed grammar relationship among pairs of words, which capture word order and relationship type (Figure 2): when considering `'nsubj'` (subject of an action) and `'dobj'` (object of an action) labels, these can be seen as grounded action-object predicates.

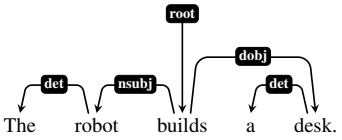


Fig. 2: Example of words that compose a sentence instance and their typed dependencies (illustrated as labeled directed edges).

We can then perform learning on such grounded models thanks to MLN, which is a knowledge representation formalism that enables probabilistic learning and inference via the combined use of first order logic and probabilistic undirected graphical models (i.e. *Markov Random Fields*). More formally, MLN theory defines a probability over the world x as a log-linear model in which we have an exponentiated sum of weights w_j of a binary feature f_j, and the partition function Z (see Equation 4).

$$P\left(X = x\right) = \frac{1}{Z} * exp\left(\sum_{j} w_j f_j\left(x\right)\right) \tag{4}$$

In our case, we consider the binary formula $f_j(x)$ as an evaluation of a logic formula representing grammar relations as predicates, and we substitute such term with $n_j(x)$, where the latter is number of true groundings of such formula f_j in x_j. The MLN formalism aims to learn the stationary distribution of the true groundings $n_j(x)$, possibly a sufficient heuristic condition for scalability.

III. Discussion

a) Related Work: Systems which focus on the initialization parameters from ontologies (i.e. aggregates of semantic relationships and entities) do not debate how such source was populated [7]. Some previous literature does value mappings between language constructs and affordances, but analyze the opposite problem [8]. Closer work which adopts MLN and grammar features has been proven successful for mining natural language instructions for the robotics domain [9], but does not focus on affordance understanding and concentrates on inferring likely action roles, while other literature does make use of MLN but does not employ grammar feature analysis [10]. Closer work does consider typed dependency extraction for semantic characterization, but does not focus on SVO tuple analysis [11], [12].

b) Evaluation: As the system can process a high number of noun-action relationships, we require an equally well populated ontology representing ground truth references. For activity and passivity labels, the scope might require manual annotation.

c) Potential: Other than fulfilling the requirement of providing an initial affordance world estimate, it can provide understanding of hidden or partially observable affordances [13], particularly useful when objects are not in full reach of the perception array. The vector space induction enables the use high-dimensional tensor computations for semantic characterization adopted in linguistics (such as compositionality and retrieval of neighbouring entries [14]), to a yet unknown extent of effectiveness within the context.

d) Limitations: Although we assume the text is confined to a technical domain, the authors of the source might make use of partly figurative wordings. As a result, the word frequency distribution would present bias or outliers (i.e. presence of erroneous co-occurrences of analyzed nouns or figurative nouns unrelated to known entities). Furthermore, also the independent word frequency of occurrence does not provide information regarding entity existence, and would require a form of normalization.

e) Conclusions: The linguistic and computational obstacles towards model effectiveness are manifold, and surely require the development of processes such as bias removal and outlier detection. However, this concept paper highlights the usage of technical text mining for affordancies acquisition, and mainly points to the potential of induced vector spaces for retrieving objects with similar affordance, or the affordance of aggregates, and above all its practical use as initial world affordance estimate.

References

[1] T. E. Horton, A. Chakraborty, and R. St. Amant, "Affordances for robots: a brief survey," *AVANT. Pismo Awangardy Filozoficzno-Naukowej*, no. 2, pp. 70–84, 2012.

[2] Z. S. Harris, "Distributional structure." *Word*, 1954.

[3] M. Sahlgren, "The distributional hypothesis," *Italian Journal of Linguistics*, vol. 20, no. 1, pp. 33–54, 2008.

[4] L. Reeve and H. Han, "Survey of semantic annotation platforms," in *Proceedings of the 2005 ACM symposium on Applied computing*. ACM, 2005, pp. 1634–1638.

[5] M. Richardson and P. Domingos, "Markov logic networks," *Machine learning*, vol. 62, no. 1-2, pp. 107–136, 2006.

[6] M.-C. de Marneffe and C. D. Manning, "The stanford typed dependencies representation," in *22nd International Conference on Computational Linguistics*, 2008, p. 1.

[7] S. S. Hidayat, B. K. Kim, and K. Ohba, "Learning affordance for semantic robots using ontology approach," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 2630–2636.

[8] O. Yürüten, K. F. Uyanık, Y. Çalışkan, A. K. Bozcuoğlu, E. Şahin, and S. Kalkan, "Learning adjectives and nouns from affordances on the icub humanoid robot," in *From Animals to Animats 12*. Springer, 2012, pp. 330–340.

[9] D. Nyga and M. Beetz, "Everything robots always wanted to know about housework (but were afraid to ask)," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 243–250.

[10] I. Beltagy, C. Chau, G. Boleda, D. Garrette, K. Erk, and R. Mooney, "Montague meets markov: Deep semantics with probabilistic logical form," in *2nd Joint Conference on Lexical and Computational Semantics: Proceeding of the Main Conference and the Shared Task, Atlanta*, 2013, pp. 11–21.

[11] S. Padó and M. Lapata, "Dependency-based construction of semantic space models," *Computational Linguistics*, vol. 33, no. 2, pp. 161–199, 2007.

[12] G. Boella, L. Di Caro, and L. Robaldo, "Semantic relation extraction from legislative text using generalized syntactic dependencies and support vector machines," *Theory, Practice, and Applications of Rules on the Web*, p. 218.

[13] W. W. Gaver, "Technology affordances," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 1991, pp. 79–84.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

Prior Affordance Understanding with Relational Learning for Human Safe Action Planning

Vishal A. Bhalla[†1] and Nicholas H. Kirk[†2]

***Abstract*— This paper discusses how to achieve scalable affordance mining from human-written texts. Relational learning performed on Subject-Verb-Object (SVO) tuples, generated from grammar analysis, provides artificial assistants with object affordances, and therefore with common sense knowledge regarding action executability. Experiments were performed on web crawled data from the database *www.wikihow.com* with the Markov Logic Network (MLN) framework Tuffy, for use in the daily household tasks domain. We showed the success of the approach by comparing results to annotated videos of the CAD120 dataset, demonstrating that affordance priors are extracted effectively.**

I. INTRODUCTION

Research in cognitive robotics envisages to impart knowledge to robots from information sources created and used by humans, so that they can get the know-how to perform daily activities. In this paper, we concern ourselves with high-level, affordance information [1] which consider active and passive labels of an entity, helping the understanding of how a particular object can be acted upon [2]. Human-written texts are a rich source for gathering an initial estimate of such action-object relations by using a probability distribution over natural language verb-noun pairs. In this work, we provide means to understand if given an entity is more likely to be in an active or passive role by mining subject-verb and object-verb relationships extracted from text, by means of logico-statistical analysis via Markov Logic Networks (MLN). As affordances are usually obtained by sensorimotor exploration with reinforcement learning, the presented approach is more safe in human robot interaction terms, because such probabilistic distribution over objects is known at initialization time.

For example, consider the text crawled from a website like *wikihow.com* as shown in Fig. 1. The website contains text with a set of instructions as performed by humans, for example describing the daily task of cleaning a floor. This text is parsed to extract grammar relationships, i.e. dependencies in each sentence. Consider the sentence *"Follow up dusting by cleaning the floor."* The dependencies parsed from it would include *dobj(cleaning-5, floor-7)*, among other relations, which clarifies that there is a semantic text relationship between "*floor*" (i.e. the *direct object*), and "*cleaning*" (i.e. the executed action). In particular, in this paper we are interested only in the Subject-Verb-Object dependencies

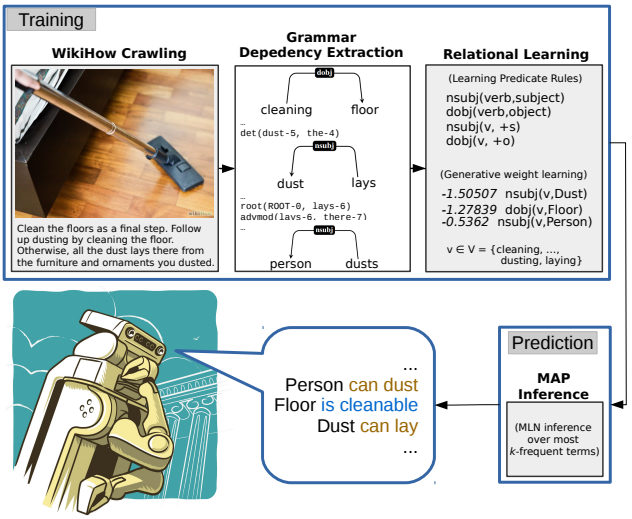


Fig. 1. System's pipeline for affordance mining from general texts: web-crawled data is parsed for subject and object grammar dependencies, which are used in a statistical relational learning framework to understand what action can be *executed by* or *executed on* a given object.

as they define the ability modality in each sentence [3]: We filter out only NSUBJ (*noun subject*) and DOBJ (*direct object*) type of dependencies, e.g. *dobj(cleaning-5, floor-7)* and *nusbj(you-13, dusts-14)*, from all the grammar relations. By gathering such tuples for all the web crawled text we can statically learn two distributions over action-entity pairs which represent the passive or active nature of the entity. In this work, via statistical relational learning we learn weights that help to infer probabilities representing each tuple occurrence, which then can be used to estimate the most likely action over an object. For example, the robot may acquire the knowledge that *"Person can dust"* and *"Floor is cleanable"*, useful for anchoring high-level entities to actions, as discussed in previous literature [4]. An illustration of the described pipeline is presented in Fig. 1.

This paper discusses in detail the approach and implementation within the artificial cognition domain, for acquiring probabilities on object affordance at large scales for each action, in view of supplying these as prior knowledge for action planning.

II. RELATED WORK

The known literature making use of mappings between affordances and language constructs solves the opposite problem at hand, i.e., uses the mapping to derive language constructs like nouns and adjectives from affordance labels

† Joint first authorship
[1] Department of Informatics, Technical University of Munich, Germany vishal.bhalla@tum.de
[2] Fraunhofer Institute for Communication Systems ESK, Germany nicholas.kirk@esk.fraunhofer.de

[5]. Another paper highlights an ontology-based affordance concept for ubiquitous robots [6], aggregating semantic relationships and entities as parameters from ontologies. However, it does not debate on how the source was populated, i.e. the focus of this contribution. Previous linguistic work oriented towards semantic characterization and typed dependency analysis create abstract textual representations (e.g. vector spaces) [7], [8], but do not focus on verb-noun tuple analysis for mining affordances, and do not apply it to the robotics domain. There is other literature which combines logical and distributional representations of natural language by transforming distributional similarity judgments into weighted inference rules using MLN [9], however without grammar feature analysis.

Another paper [10] highlights the structured nature of affordances, and suggests probabilistic ontologies based on MLN to model and infer object-action relationships, but focuses on inference of missing affordances and does not consider population from grammar tuples. A previous concept paper [3] highlights the possible use of technical text mining for affordance acquisition, but this work nor others discuss large scale statistical relational learning, nor its evaluation for affordance mining. The closest work focusing on MLN, language and artificial cognition together uses text to infer action roles for action plan disambiguation [11]. However, such work does not focus on the scale of the text mining, nor on the application of affordances, nor on the informativeness analysis per se. In comparison to this, our work achieves scalability by reducing the logical learning rule set and by using the scalable system Tuffy [12], which scales up weight learning and statistical inference in MLN by using a Relational DataBase Management System (RDBMS).

III. MODEL

We now present the theoretical foundations of the main two technologies we adopt: Stanford typed Dependencies (SD) for the grammatical analysis, and the statistical relational framework of Markov Logic Networks (MLN), which in our scope learns the probabilistic distributions over grammar instances.

A. Stanford Typed Dependencies

This paper makes use of the Stanford typed dependencies (SD) representation, which was designed to provide a straightforward description of grammatical relations for any user who could benefit from automatic text understanding [13]. SDs have a simple design that provides semantically meaningful information as well as an automatic procedure to extract the relations. The SD representation for the example sentence [14] "*Bell, based in Los Angeles, makes and distributes electronic, computer and building products.*" is as follows:

nsubj(makes-8, Bell-1)
nsubj(distributes-10, Bell-1)
vmod(Bell-1, based-3)
nn(Angeles-6, Los-5)
prep in(based-3, Angeles-6)
root(ROOT-0, makes-8)
conj and(makes-8, distributes-10)
amod(products-16, electronic-11)
conj and(electronic-11, computer-13)
amod(products-16, computer-13)
conj and(electronic-11, building-15)
amod(products-16, building-15)
dobj(makes-8, products-16)
dobj(distributes-10, products-16)

The format specifies the type of the relation, and the formal arguments of such predicate with ordinal numbers for co-referencing. Out of the many SD types, we select only NSUBJ and DOBJ for our proposed solution as they reliably represent activity and passivity of an entity, and present lower parsing error rates compared to other more complex grammatical structures.

B. Markov Logic Networks (MLN)

A Markov logic network MLN makes use of a probability distribution function of undirected graphical models (i.e. Markov Random Fields) and first order logic as knowledge representation formalism. MLNs combine statistical and logical reasoning, and are now emerging as a powerful framework which is being used in many data intensive problems including information extraction, entity resolution, and text mining. First Order Logic (FO) is defined by a set of predicates quantified by existence or universality, and is the level of logical abstraction that MLN makes use of.

More precisely, MLN theory defines a probability over the world x as a log-linear model in which we have an exponentiated sum of weights w_j of a binary feature f_j, and the partition function Z (see Eq. 1).

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_j w_j f_j(x)\right) \tag{1}$$

In our case, we consider the binary formula $f_j(x)$ as an evaluation of a logic formula representing grammar relations as predicates, and we substitute such term with $nj(x)$, where the latter is number of true groundings of such formula f_j in x_j. The MLN formalism aims to learn the stationary distribution (i.e. learn stationary weight values w_j) of the true groundings $n_j(x)$, possibly a sufficient heuristic condition for scalability. Inference involves finding the most probable state of a grounded MLN given some evidence, and is thus an instance of weighted satisfiability. The reader might debate why MLN, a framework with higher computational complexity compared to other learning formalisms, is adopted in this work. Naive Bayes methods on grammatical relationships (as implemented in [15]), for example, do not directly model the underlying semantic structure of grammatical relationships, and does not enable inference, which MLN caters for.

Current implementations of MLNs do not scale to large real-world data sets, which is preventing their widespread adoption. For this reason we use *Tuffy* [12], that achieves scalability via three novel contributions:

1) Maximum use of the relational optimizer using a bottom-up approach to grounding,
2) a novel hybrid architecture that uses an RDBMS for queries, and

3) Builds novel partitioning, loading, and parallel algorithms that improve the efficiency of the stochastic local search.

IV. SYSTEM FLOW

The system takes as input human-written text, to then i) first parse it to obtain grammatical tuples. The parser is built using grammar rules on SD and extracts semantic dependencies in the text. However, as we are interested only in SVO tuples, we ii) make use of a custom *filter* component. The filter strips the position numbers for each word and selects only those verb-noun pairs which will help us make an informed decision regarding affordances (i.e. Subject-Verb (NSUBJ) and Object-Verb (DOBJ) tuples). These filtered tuples are then used to construct potentiality relationships by iii) supplying such evidence as training data to the MLN framework. We now provide further information regarding the logical rules we adopt for the learning and inference phase.

A. Implementation

In our MLN setting, we learn the frequency distribution for each tuple occurrence for specified predicate rules, which expand and ground (i.e. instantiate) only the entity (subject or object). More precisely:

dobj(ve,+obj)

nsubj(ve,+subj)

The '+' operator indicates that the formula is expanded with respect to its corresponding evidence values. As example, for the formula used in our domain, +subj indicates that a rule for each subject entity will be created in the database. Conversely after training (i.e. weight learning), we used an empty database and marginal inference (which employs MC-SAT, an MCMC inference algorithm) to estimate and infer the marginal probabilities. A full description of our implemented pipeline is described in Algorithm 1, and exemplified in Figure 1.

Algorithm 1 Object Affordance Learning

procedure AFFORDANCELEARNING()
Web Crawling:
  for each URL in $URLList$ **do**
    text ← text + WebCrawlText(URL)
    // Web crawl text from the given link
Dependency Parsing:
  for each $sentence$ in $text$ **do**
    URL ← StanfordDependencyParsing(sentence)
    // Parse text to get semantic relations
Filter:
  for each $tuple$ in $grammartuples$ **do**
    SVOtuple ← FilterSVOTuples(tuple)
    // Filter out only nsubj and dobj SD
    formatTuple ← FormatSVOTuples(SVOtuple)
    // Remove position labels
    // Perform stemming & lemmatization on words
Ability Modality Processing:
  learnwts ← MLNWeightLearning(rules, evidence, query)
  // Learn weights from the frequency distribution of all tuples as evidence
  inferProb ← MLNInference(learnwts, noEvidence, query)
  // Infer a probabilistic estimate of the SV & VO tuples

V. EVALUATION CRITERIA

Given the now explained pipeline procedure (exemplified in Figure 1), we proceed to evaluate by comparing the priors obtained with our system, with ground truths we obtained by manually annotating the videos from the everyday activity dataset CAD 120 [16], on a $i5$@2.4GHz 64-bit machine with 4 GB of process-dedicated RAM.

A. Training Phase

We require precise potentiality relationships which can be extracted from instruction manual-like text. In order to ensure maximum word coverage, we pick how-to descriptions from assorted domains ranging from kitchen utensils to furniture building, from the WikiHow repository (*www.wikihow.com*). WikiHow is an optimal source for instruction based text. We performed text crawling on 100 different instructions and processed them with the presented pipeline. As dependencies (SD) are sentence bound, this text is iteratively parsed sentence by sentence to get all relationships. In quantitative terms, for 46 links, the total number of nsubj and dobj tuples in this crawled text were 1007 and 950, respectively.

B. Testing Phase

As ground truth evaluation we use the CAD-120 dataset [16] comprising of RGB-D video sequences of daily human activities. We are interested in using the 10 high-level activities: MAKING CEREAL, TAKING MEDICINE, STACKING OBJECTS, UNSTACKING OBJECTS, MICROWAVING FOOD, PICKING OBJECTS, CLEANING OBJECTS, TAKING FOOD, ARRANGING OBJECTS AND HAVING A MEAL. The annotated video labels give us the precise information on the SVO tuples in consideration.

We represent these interactions in the form of nsubj(verb, noun1) and dobj(verb, noun2) where the common 'verb' is the action from a subject 'noun1' to an object 'noun2'. As for the pipeline itself, the filter component processes the ground truth to perform stemming using Porter's Algorithm [17]. Also, word lemmatization is done on each tuple [18] to get its root forms with verb or noun as appropriate contexts. This widely used technique for normalizing words helps to get

Link #	Word #	nsubj #	dobj #	CC %	MC %	OoV %
1	550	34	32	6	2	92
2	736	46	47	8	4	88
3	1333	83	91	16	0	84
5	1277	87	78	14	0	86
10	3157	204	208	30	2	68
15	4619	312	293	44	6	50
20	6613	454	420	46	8	46
30	9853	658	640	40	18	42
46	15078	1007	950	50	20	30
100	31212	2087	1986	80	0	20

TABLE I

RESULTS OF THE SYSTEM'S EVALUATION FOR DIFFERENT AMOUNTS OF LINKS SHOWING WORD, SUBJECT AND OBJECT CARDINALITY, AS WELL AS CLASSIFICATION RESULTS AGAINST THE ANNOTATED CAD120 DATASET (CC FOR CORRECTLY CLASSIFIED, MC FOR MISCLASSIFIED, AND OOV FOR OUT OF VOCABULARY).

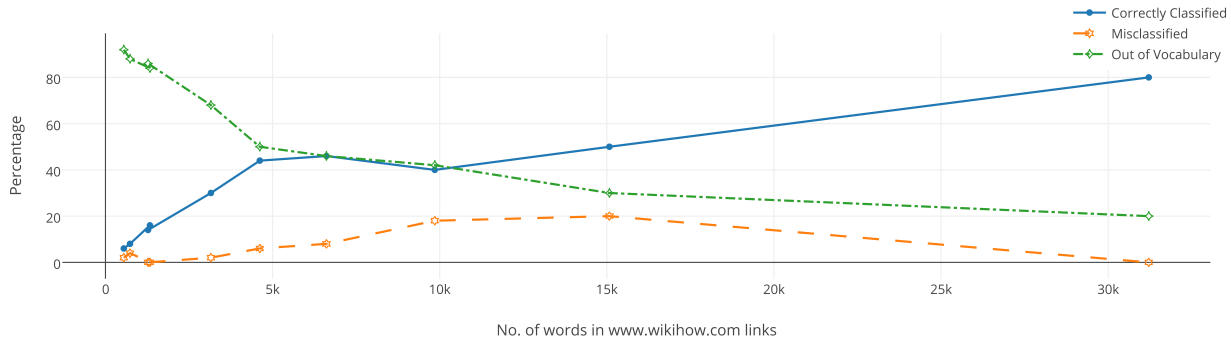


Fig. 2. Evaluation in terms of classification rates for the different orders of word magnitudes of the crawled dataset.

inflected forms of all words to their root forms and facilitates comparison.

For the the CAD-120 test corpora we have in total 1806 distinct constructed *nsubj* and *dobj* predicates. The total number of distinct objects in test set is 10.

A conceptual evaluation is, that we do not attribute as insecure something that is safe, nor the inverse, which can be evaluated in terms of object-subject misclassification. Another useful interpretation of the capability of this system is in terms of scalability, which can be evaluated in terms of amount of processed text. As such, a largely scaled system should ensure maximum word coverage, so that out-of-vocabulary exceptions are minimized, i.e. the affordance database would be more complete and useful in uncommon scenarios. We performed evaluation by randomly selecting subsets of 1,2,3,5,10,15,20,30 from 46 WikiHow links (with exception for the 46 and 100-link entries), and averaging results over 5 different iterations. In Fig. 2 and Tab. I we provide the values processed by our pipeline in terms of predicate evidence quantity and subsequent classification results. The system shows optimal convergence rates of entity coverage and acceptable classification rates for a relatively small dataset.

VI. CONCLUSIONS

In this work, we implemented a scalable system to extract object affordance priors from human-written texts for safe human robot interaction. The basic assumption is that subject-verb-object (SVO) tuples extracted from $WikiHow$ instructions have the information to model potentiality relationships over entities. Currently implemented with Stanford Typed Dependencies and Markov Logic Networks (MLN), this system can be easily scaled to incorporate a larger word corpus (we successfully tested 100 links). An extensive evaluation against an annotated version of the CAD120 dataset showed the success of the proof of concept. Future investigations will concentrate on efficiency comparison with naive Bayes methods, as well as increasing classification rates by increasing the ruleset, which would exploit better the underlying semantic structure of texts.

REFERENCES

[1] J. J. Gibson, "The theory of affordances," *Hilldale, USA*, 1977.

[2] D. A. Norman, *The psychology of everyday things*. Basic books, 1988.

[3] N. H. Kirk, "Towards learning object affordance priors from technical texts," in *"Active Learning in Robotics" Workshop, IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2014.

[4] M. Tenorth, D. Nyga, and M. Beetz, "Understanding and executing instructions for everyday manipulation tasks from the world wide web," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1486–1491.

[5] O. Yürüten, K. F. Uyanık, Y. Çalışkan, A. K. Bozcuoğlu, E. Şahin, and S. Kalkan, "Learning adjectives and nouns from affordances on the icub humanoid robot," in *From Animals to Animats 12*. Springer, 2012, pp. 330–340.

[6] S. S. Hidayat, B. K. Kim, and K. Ohba, "Learning affordance for semantic robots using ontology approach," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 2630–2636.

[7] S. Padó and M. Lapata, "Dependency-based construction of semantic space models," *Computational Linguistics*, vol. 33, no. 2, pp. 161–199, 2007.

[8] G. Boella, L. Di Caro, and L. Robaldo, "Semantic relation extraction from legislative text using generalized syntactic dependencies and support vector machines," in *Theory, practice, and applications of rules on the web*. Springer, 2013, pp. 218–225.

[9] I. Beltagy, C. Chau, G. Boleda, D. Garrette, K. Erk, and R. Mooney, "Montague meets markov: Deep semantics with probabilistic logical form," *Proceedings of* SEM*, pp. 11–21, 2013.

[10] Y. Zhu, A. Fathi, and L. Fei-Fei, "Reasoning about object affordances in a knowledge base representation," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 408–424.

[11] D. Nyga and M. Beetz, "Everything robots always wanted to know about housework (but were afraid to ask)," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 243–250.

[12] F. Niu, C. Ré, A. Doan, and J. Shavlik, "Tuffy: Scaling up statistical inference in markov logic networks using an rdbms," *Proceedings of the VLDB Endowment*, vol. 4, no. 6, pp. 373–384, 2011.

[13] M.-C. De Marneffe and C. D. Manning, "The stanford typed dependencies representation," in *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*. Association for Computational Linguistics, 2008, pp. 1–8.

[14] ——, "Stanford typed dependencies manual," Technical report, Stanford University, Tech. Rep., 2008.

[15] N. H. Kirk, G. Zhang, and G. Groh, "Estimating grammar correctness for a priori estimation of machine translation post-editing effort," in *Proceedings of the EACL 2014 Workshop on Humans and Computer-assisted Translation, Stockholm, Sweden*. Association for Computational Linguistics, 2014, pp. 16–21.

[16] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013.

[17] S. Bird, "Nltk: the natural language toolkit," in *Proceedings of the COLING/ACL on Interactive presentation sessions*. Association for Computational Linguistics, 2006, pp. 69–72.

[18] M. Toman, R. Tesar, and K. Jezek, "Influence of word normalization on text classification," *Proceedings of InSciT*, pp. 354–358, 2006.

Chapter 4

Conclusions

4.1 Concluding remarks

Artificial embodied assistants, in order to rationally act, infer inductively the necessary information for action on the basis of their sensory perception. Such information is simultaneously available in large amount but always insufficient in qualitative terms, so there is the need to i) understand what data is redundant when recognizing or storing an action abstraction, ii) infer the unknown partial knowledge necessary for decision making. The creation of symbolic data (i.e. fragments of generalized knowledge) and its parametrization is often dependent on external "common sense" knowledge, usually referring to context or frequently occurring associations. The present thesis for consideration of the "MPhil by publication" provides a contribution in such area of neurosymbolic reasoning, by debating the *source of symbols* and their manipulation (i.e. from natural language or from observation), their *representation* (i.e. action symbols based on human motor coordination principles), as well as their *disambiguation*, for instance when a natural language instruction from a human is excessively underspecified. The contributions here presented provided independent results published in different

venues, and were brought together in a single scope of symbolic reasoning applied to *context analysis and generalization, to achieve context-independence*.

4.2 Future work

On the road towards technical cognitive systems, *prospection*, i.e. the act of internal simulation and evaluation of possible future events, is of paramount importance to implement decision making, where a decision is the choice of the incoming action to execute given the past executions and local knowledge. These two problems are usually enacted with two different theoretical frameworks: the first is known in artificial intelligence theory as *induction* (Solomonoff 1964), where the prediction system forecast considers the history of executed actions and the complexity of the action hypotheses (Solomonoff 1964; Li and Vitányi 2013), while the second is the use of partial knowledge for *inference* problems to deduce the most likely world.

To extend the logico-statistical representation systems presented in this thesis, future work will explore computable Bayesian induction models, enabling i) the intrinsic complexity selection of the "automatic Occam's Razor" (Jefferys and Berger 1992; Rasmussen and Ghahramani 2001), combining also ii) inference from partial evidence, to exploit information from sensor and memory models. This can be seen as the *prediction of action-object pairs (such as Object-Action Complexes (OAC)*, (Kraft et al. 2008)) *within a sequence*, on the basis of sensory information, adapting the likelihood distribution over time on the basis of the context. This, in the view of the author, is a major necessary contribution towards autonomy: Given the possibility of uniquely associating a perception component feed to a reliability estimate, the author believes that the reliability of these can be modeled via reinforcement learning, enabling the learning of higher-level behavior policies. An example of this is, for instance, is to take a decision based on multiple likelihood

contributors (as in Kirk et al. 2015), and understanding over time, in an online fashion, which of these sources are higher contributors to the decision making than others, on the basis of context. While previous systems understand the importance of Bayesian inference in cognitive decision making (e.g. many important neuroscientific pieces of work Knill and Richards 1996; Rao 2004), and of (voted) classifier aggregation (Cho and Kim 1995; Kittler 1998), they do not perform online learning of the weight of the contribution to the decision making. In other terms, none of the known previous works performs *classifier fusion for online reasoning in object-action perception-execution contexts*. Current state-of-the-art systems, given the "narrowness" of today's neural and reasoning applications, therefore require a predictor aggregation schema. The author claims that such reliability estimates can model the degree to which the present contextual information may present evidence towards *a fortiori* reasoning, e.g. evidence which implies a strong argument, which in turn provides information towards minor arguments. The simultaneous learning of multiple reliability estimates which contribute to a given decision, are key to different hypothetico-deductive reasoning capabilities, and allegedly many other higher-cognitive reasoning capabilities. This will be subject of future thorough investigation by modeling reinforcement policies.

Bibliography

Anderson, John, and Mark Evans. 1996. 'Constraint-directed improvisation for complex domains.' In *Conference of the Canadian Society for Computational Studies of Intelligence*, 1–13. Springer.

Arora, Sanjeev, and Boaz Barak. 2009. *Computational complexity: a modern approach.* Cambridge University Press.

Beetz, Michael, Ferenc Bálint-Benczédi, Nico Blodow, et al. 2015a. 'Robosherlock: Unstructured information processing for robot perception.' In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 1549–1556. IEEE.

Beetz, Michael, Moritz Tenorth, and Jan Winkler. 2015b. 'Open-EASE.' In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 1983–1990. IEEE.

Bhalla, Vishal A, and Nicholas H Kirk. 2016. 'Prior Affordance Understanding with Relational Learning for Human Safe Action Planning.' In *2016 IEEE International Conference on Robotics and Automation (ICRA), Workshop on AI for Long-Term Autonomy, At Stockholm, Sweden.* IEEE.

Cangelosi, Angelo, Alberto Greco, and Stevan Harnad. 2000. 'From robotic toil to symbolic theft: Grounding transfer from entry-level to higher-level categories1.' *Connection Science* 12 (2): 143–162.

Cangelosi, Angelo, Emmanouil Hourdakis, and Vadim Tikhanoff. 2006. 'Language acquisition and symbol grounding transfer with neural networks and cognitive robots.' In *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, 1576–1582. IEEE.

Cho, Sung-Bae, and Jin H Kim. 1995. 'Combining multiple neural networks by fuzzy integral for robust classification.' *IEEE Transactions on Systems, Man, and Cybernetics* 25 (2): 380–384.

Coradeschi, Silvia, and Alessandro Saffiotti. 2003. 'An introduction to the anchoring problem.' *Robotics and Autonomous Systems* 43 (2): 85–96.

Dounskaia, Natalia. 2005. 'The internal model and the leading joint hypothesis: implications for control of multi-joint movements.' *Experimental Brain Research* 166 (1): 1–16.

Falco, Pietro, Matteo Saveriano, Eka Gibran Hasany, Nicholas H Kirk, and Dongheui Lee. 2017. 'A Human Action Descriptor Based on Motion Coordination.' *IEEE Robotics and Automation Letters* 2 (2): 811–818.

Ficuciello, Fanny, Pietro Falco, and Sylvain Calinon. 2018. 'A Brief Survey on the Role of Dimensionality Reduction in Manipulation Learning and Control.' *IEEE Robotics and Automation Letters*.

Harris, Zellig S. 1954. 'Distributional structure.' *Word* 10 (2-3): 146–162.

Hinton, Geoffrey E, and Ruslan R Salakhutdinov. 2006. 'Reducing the dimensionality of data with neural networks.' *science* 313 (5786): 504–507.

Jefferys, William H, and James O Berger. 1992. 'Ockham's razor and Bayesian analysis.' *American Scientist* 80 (1): 64–72.

Kaltenbacher, Simon, Nicholas H Kirk, and Dongheui Lee. 2015. 'A Preliminary Study on the Learning Informativeness of Data Subsets.' In *The 8th International Workshop on Human-Friendly Robotics (HFR 2015), Munich, Germany*.

Kirk, Nicholas H. 2014. 'Towards Learning Object Affordance Priors from Technical Texts.' In *"Active Learning in Robotics" Workshop, 2014 IEEE-RAS International Conference on Humanoid Robots.* IEEE.

Kirk, Nicholas H., Daniel Nyga, and Michael Beetz. 2014. 'Controlled Natural Languages for Language Generation in Artificial Cognition.' In *2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China*, 6667–6672. IEEE RAS.

Kirk, Nicholas Hubert, Karinne Ramirez-Amaro, Emmanuel Dean-Leon, Matteo Saveriano, and Gordon Cheng. 2015. 'Online Prediction of Activities with Structure: Exploiting Contextual Associations and Sequences.' In *2015 IEEE-RAS International Conference on Humanoid Robots, Seoul, Korea.* IEEE.

Kittler, Josef. 1998. 'Combining classifiers: A theoretical framework.' *Pattern analysis and Applications* 1 (1): 18–27.

Knill, David C, and Whitman Richards. 1996. *Perception as Bayesian inference.* Cambridge University Press.

Kok, Stanley, and Pedro Domingos. 2005. 'Learning the structure of Markov logic networks.' In *Proceedings of the 22nd international conference on Machine learning*, 441–448. ACM.

Kraft, Dirk, Nicolas Pugeault, EMRE BAŞESKI, et al. 2008. 'Birth of the object: Detection of objectness and extraction of object shape through object–action complexes.' *International Journal of Humanoid Robotics* 5 (02): 247–265.

Krüger, Norbert, Christopher Geib, Justus Piater, et al. 2011. 'Object–action complexes: Grounded abstractions of sensory–motor processes.' *Robotics and Autonomous Systems* 59 (10): 740–757.

Kulic, Dana, Dongheui Lee, Christian Ott, and Yoshihiko Nakamura. 2008. 'Incremental learning of full body motion primitives for humanoid robots.' In *Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on*, 326–332. IEEE.

Kulic, Dana, Wataru Takano, and Yoshihiko Nakamura. 2009. 'Online segmentation and clustering from continuous observation of whole body motions.' *IEEE Transactions on Robotics* 25 (5): 1158–1166.

Latash, Mark L. 2010. 'Motor synergies and the equilibrium-point hypothesis.' *Motor control* 14 (3): 294–322.

Latash, Mark L, John P Scholz, and Gregor Schöner. 2007. 'Toward a new theory of motor synergies.' *Motor control* 11 (3): 276–308.

Lemaignan, Séverin, Raquel Ros, Lorenz Mösenlechner, Rachid Alami, and Michael Beetz. 2010. 'ORO, a knowledge management platform for cognitive architectures in robotics.' In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3548–3553. IEEE.

Li, Ming, and Paul Vitányi. 2013. *An introduction to Kolmogorov complexity and its applications.* Springer Science & Business Media.

Magnanimo, Vito, Matteo Saveriano, Silvia Rossi, and Dongheui Lee. 2014. 'A bayesian approach for task recognition and future human activity prediction.' In *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*, 726–731. IEEE.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. 'Distributed representations of words and phrases and their compositionality.' In *Advances in neural information processing systems*, 3111–3119.

Müller, M., T. Röder, M. Clausen, et al. 2007. *Documentation Mocap Database HDM05.* Technical report CG-2007-2. Universität Bonn, June.

Nyga, Daniel, and Michael Beetz. 2012. 'Everything robots always wanted to know about housework (but were afraid to ask).' In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 243–250. IEEE.

Rao, Rajesh PN. 2004. 'Bayesian computation in recurrent neural circuits.' *Neural computation* 16 (1): 1–38.

Rasmussen, Carl Edward, and Zoubin Ghahramani. 2001. 'Occam's razor.' *Advances in neural information processing systems:* 294–300.

Richardson, Matthew, and Pedro Domingos. 2006. 'Markov logic networks.' *Machine learning* 62 (1): 107–136.

Russell, Stuart Jonathan, Peter Norvig, John F Canny, Jitendra M Malik, and Douglas D Edwards. 2003. *Artificial intelligence: a modern approach.* Vol. 2. 9. Prentice hall Upper Saddle River.

Russell, Stuart, Peter Norvig, and Artificial Intelligence. 1995. 'A modern approach.' *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs* 25:27.

Saveriano, Matteo. 2017. 'Robotic Tasks Acquisition via Human Guidance: Representation, Learning and Execution.' Dissertation, Technische Universität München.

Schwitter, Rolf. 2010. 'Controlled natural languages for knowledge representation.' In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, 1113–1121. Association for Computational Linguistics.

Schwitter, Rolf, and Marc Tilbrook. 2004. 'Controlled natural language meets the semanticweb.' In *Proceedings of the Australasian Language Technology Workshop 2004*, 55–62.

Solomonoff, Ray J. 1964. 'A formal theory of inductive inference. Part I.' *Information and control* 7 (1): 1–22.

Sutton, Charles, and Andrew McCallum. 2006. *An introduction to conditional random fields for relational learning*. Vol. 2. Introduction to statistical relational learning. MIT Press.

Takano, Wataru, and Yoshihiko Nakamura. 2008. 'Integrating whole body motion primitives and natural language for humanoid robots.' In *Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on*, 708–713. IEEE.

Tenorth, Moritz, and Michael Beetz. 2009. 'KnowRob—knowledge processing for autonomous personal robots.' In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, 4261–4266. IEEE.

Thomaz, Andrea, Guy Hoffman, Maya Cakmak, et al. 2016. 'Computational human-robot interaction.' *Foundations and Trends® in Robotics* 4 (2-3): 105–223.

Wächter, Mirko, Sebastian Schulz, Tamim Asfour, et al. 2013. 'Action sequence reproduction based on automatic segmentation and object-action complexes.' In *Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on*, 189–195. IEEE.