Research Article Open Access

Laith Alkurdi*, Christian Busch, and Angelika Peer

Dynamic contextualization and comparison as the basis of biologically inspired action understanding

https://doi.org/10.1515/pjbr-2018-0003
Received February 29, 2016; accepted January 22, 2018

Abstract: People exhibit a robust ability to understand the actions of others around them. In this work, we identify two biologically inspired mechanisms that we hypothesize to be central in the function of action understanding. The first module is a contextual predictor of the observed action, given the goal-directed movement towards objects, and the actions that are allowed to be performed on the object. The second module is a kinematic trajectory parser that validates the previous prediction against a set of learned templates. We model both mechanisms and link them to the environment using the cognitive framework of Dynamic Field Theory and present our first steps into integrating the aforementioned modules into a consistent framework for the purpose of action understanding. The two modules and the combined architecture as a whole are experimentally validated using a recording of an actor performing a series of intentional actions testing the ability of the architecture to understand context and parse actions dynamically. Our initial qualitative results show that action understanding benefits from the combination of the two modules, while any module alone would be insufficient to resolve ambiguity in the perceived actions.

Keywords: dynamic field theory, action understanding, embodied embedded cognition, affordance theory, theory of mind

***Corresponding Author: Laith Alkurdi:** Chair of Automatic Control Engineering, Department of Electrical, Electronic and Computer Engineering, Technische Universität München, München, Germany, E-mail: laith.alkurdi@tum.de
Christian Busch: Chair of Automatic Control Engineering, Department of Electrical, Electronic and Computer Engineering, Technische Universität München, München, Germany
Angelika Peer: Bristol Robotics Laboratory, Faculty of Environment and Technology, Department of Engineering Design and Mathematics, University of the West of England, Bristol, England

1 Introduction

The promise of intelligent robots sharing the environment with human agents and collaborating towards a common goal has been a major driving force for assistive robotics applications [1, 2]. That in itself requires robots to be endowed with capabilities that are comparable to humans in behavior production and environmental reasoning. Human behavior and cognitive reasoning abilities can be seen as a dynamic, complex interaction between the body, brain and the environment the human agent is situated in. It is the tight coupling between the agent's sensory and motor systems and the environment that gives rise to a series of adaptive and proactive actions to fulfill a certain intention. The situated embodied view of cognition encompasses the above ideas [3]. Furthermore, it aims to include the agent's past experiences as well as the neuronal processes to its understanding of behavior and cognition [4, 5].

Action understanding (AU) can be defined as the task of classifying a stream of human-related multimodal data (motion, audio, contextual, etc.) into semantic terms suitable for influencing the future intelligent behavior to support the human agent in a meaningful manner. Intelligent systems face several challenges in AU. These include the spatiotemporal variation within a class of actions, as well as interclass and intraclass variation in how persons perform actions. The spatiotemporal variation here refers to the fact that similar actions might vary in duration and path followed across agents and trials. Another major challenge is the large search space of actions available to an agent in any environment [6, 7]. To be able to understand an action, intelligent systems need to be able to solve the spatiotemporal variation problem by a robust trajectory recognition system, and the large search space problem by incorporating the context of the action.

In our quest towards an end-to-end biologically-inspired architecture for hierarchical human action understanding, we present two systems that address the challenges mentioned above and that we hypothesize to be central to the task of AU. The two systems are inspired by processes observed within human behavioral studies, as discussed in section 2. The main challenges addressed in this work are the context understanding of an observed movement and the trajectory parsing of the movement. Additional secondary challenges addressed in this work include how the context understanding interacts with trajectory parsing, and how visual information of motion can be used as an input in a manner consistent with the complete system. The work presented is inspired by definitions within the embodied situated cognition stance, as discussed in section 2.1. The context understanding is based on definitions of affordances as given in section 2.2, and the trajectory parsing follows ideas of biological motion perception as discussed in section 2.3. The modules and the whole systems are modeled using the cognitive framework developed within dynamic field theory (DFT).

The modeling of action understanding systems using DFT has been addressed recently in the literature. A neural dynamic approach for parsing a sequence of actions was presented in [8] by Lobato et al. The authors present a neural-dynamic architecture that is capable of detecting and representing a sequence of actions, namely reaching/grasping/dropping objects on a table-top scenario. Trajectory recognition was not considered, but rather three-dimensional positions of hands and objects were used to calculate whether the hand was approaching the object or not. The overall architecture is capable of memorizing a string of actions for overall action understanding. Similar work was also presented within neural fields in the work of Bicho et al., in which the focus was on integrating verbal and nonverbal communication in a joint-assembly task in which the sequence of actions was given [9]. In contrast to the work presented by Lobato et al. and Bicho et al. we extend the application area of DNFs towards representation and recognition of temporally extended actions using context and movement information. Furthermore, while the work presented in Lobato et al. and Bicho et al. deal with only table-top scenarios, we present systems that are general enough for understanding locomotion, manipulation, and actions in free-space.

Regarding the task of AU itself, there exist many ways to understand actions an agent might perform, which renders a large search space for an AU system. We address this problem of context understanding by modeling three processes into a contextual action understanding system. Firstly, we model the detection of goal-directed movements. Secondly, we model the shifting of attention from joints (end-effector) to objects in the line of action of the joint movements. Finally, we model the context understanding of the movement given the affordances of the objects towards which the attention was shifted. The term affordances relates to the action possibilities that an object might allow [10]. A thorough definition of affordances is given in section 2.2. The context of the movement based on affordances is understood using a novel contextual action recognition system (CARS). This CARS is composed of several contextual action recognition modules (CARMs) which are further discussed in section 5.2. Several CARMs are used as one CARM is needed for every item of interest (e.g., end-effector) that we might want to track. The function of the CARS is to pick the most relevant subset of templates, in a pre-learned database of templates, that represent movement features. A separate affordance logic block aids in this selection, and is further discussed in section 5.4.

The second AU challenge addressed in this work is trajectory parsing. This online comparison is performed within the trajectory action recognition system (TARS). This system allows for spatiotemporal variation between the template and the observed motion and outputs a positive result if they are matched. The TARS is composed of several action recognition modules (TARMs) specific for each action to be recognized as discussed in section 5.3. The transformation from the visual input of joint movements into biologically-inspired features for comparison purposes is considered and further discussed in section 5.1.

Overall, the AU architecture in this work presents, for the first time, a novel predictive system within DFT that models attention-shifts and pairs up with a trajectory parsing system in a second step. The trajectory parsing system takes account of spatial as well as temporal variations that are usually problematic when understanding actions. Particular attention is given to how objects and the environment are integrated into the overall architecture and on how they can drive action understanding. The two modules and the combined architecture as a whole are experimentally validated using a recording of an actor performing a series of intentional actions. This experiment focuses on the ability of the architecture to understand the context and parse actions dynamically. Our initial qualitative results, which are given in section 6, show that action understanding benefits from the combination of the two modules, while any module alone would be insufficient to resolve ambiguity in the perceived actions. A complete discussion of the results and the AUA itself is given in section 8.

Compared to the state-of-the-art, the AU architecture in this work combines both context recognition and trajectory recognition rather than opting for either contextual recognition alone or trajectory parsing by itself for the task of action understanding. Furthermore compared to the related work within DFT we explicitly model objects and their affordances in a manner that is consistent with definitions in the situated, embodied view of cognition that DFT is built upon. The application domain of this model ranges from scenario understanding to human-robotic interaction scenarios where intelligent systems are expected to assist humans in a meaningful manner. [1, 2]. The model's strength stems from the interaction between the contextual systems (CARS), the trajectory parsing system (TARS) and the affordance system, such that a wide range of actions (manipulation, locomotion and free-space actions) could be understood. The model suffers from a few limitations currently. Firstly, the model makes use of a few algorithmic shortcuts that are not biologically plausible. Secondly, the current technical implementation is restrictive (e.g., due to slow offline template generation). A faster implementation would be a topic for future work such that a full evaluation of system performance across different scenarios becomes possible.

2 Background

In this section we aim at defining the main concepts that motivate our cognitive action understanding approach formally. We will discuss the ideas behind situated cognition in section 2.1, and how it motivates the concept of affordances that is further discussed in section 2.2. We also give a brief discussion on human movement perception in section 2.3 and discuss how it has been historically significant to the problem of action understanding. Finally, in section 2.4, we discuss the connections of the systems developed in this work to findings in neuroscience.

2.1 Situated Embodied Embedded Cognition

The basic hypothesis behind situated cognition is that behavior is a product of the dynamic interaction between the agent and its environment, and is inseparable from the context that it emerges from [4, 11, 12]. Information is thought to be a product of the coupling between the agent and its environment rather than an a priori representation in the agent's brain as proposed by traditional views of cognition. Situated cognition shares ideas with ecological psychology [10] and intentional dynamics [13]. Moreover, cognition, as defined here, is understood as a continuous state in which motor-sensory systems interact dynamically and thus can be described naturally using ideas from dynamic system theory [5].

We define action understanding to be a dynamic process that respects the tightly coupled interaction between the motor system, sensory information, and the environment. We use DFT to model this dynamic process as DFT provides the required tools, e.g., the stability of attractor states, localized-bump representations, etc. to model the, e.g., link between environment and sensors. We discuss DFT in detail in section 3. Situated cognition shares ideas with the field of ecological psychology, specifically with definitions of affordances which we discuss further in the following section.

2.2 Affordances

There has been accumulating evidence that actions are coded in their goals [14–17]. Direct perception of action possibilities (affordances) of the objects available in the environment and the goal-directed movements towards them could give a hint of what the context of the action is [10, 18]. The CARS presented in this work models this process. The function of the CARS is to understand goal-directed behaviors through prediction of the object to be manipulated and the processing of the affordances of that object.

The term affordances was introduced by Gibson as a general concept to explain what the environment can afford for an agent, and what the action possibilities are [10, 18, 19]. The exact definition of affordances has been a point of dispute since it was introduced by Gibson himself, leading to a range of attempts to formalize the concept [20–24]. In this work, however, we take inspiration from the previous references and define affordances as agent-relative, activity-potentials an agent directly perceives from the immediate environment. They are agent-relative in the sense that the affordances are attributed to environmental objects with respect to agent parameters (e.g height, width, ability, etc.), as an example, an infant's chair might not afford sitting on for an adult and so on [25]. Trajectory information should also be used alongside context to validate the results of the contextual information and solve any ambiguity when several possible affordances/contexts are present. In the following, we discuss what biological motion perception is, and how trajectories are perceived and understood biologically.

2.3 Biological motion perception

The early works of Jules Marey [26] and Gunnar Johansson [27] tried to study human biological motion perception by attaching markers or light sources to the joints of a human in a dark suit. The animation produced when recording the activity of the actor is known as point light animation (PLA). These PLAs were of great interest as humans were successful at recognizing the underlying action when the animations were shown to them [28]. These results indicate that stored patterns of movement information could be used to interpret incoming sensory information of movement. Indeed, biological systems depend on a stream of features (stimulus) produced by static views of the body to perceive and classify movement patterns [29]. These features can be thought as form cues of a specific body configuration, to that end Giese in [30] discusses the concept of snapshots to explain how biological motion could be solved. We direct the reader to the Giese's work in [31–34] for further discussion. Therefore, information extracted from biological motion is crucial to the understanding of human movements [35, 36]. The TARS within our work models these ideas and aims at understanding actions by parsing the movement trajectories.

2.4 Relation to Neuroscience

Action and context understanding is also observed on a neuronal level in biological agents, e.g., as functions of the mirror neuron system (MN) and the Canonical neuron system (CN), respectively. MNs are specific neurons in the agent's brain that fire not only when the agent is performing an action, but also when the same goal-directed action is observed. Their proposed function is to represent an embodied process that allows action and intention recognition [14, 37] as well as Theory of Mind [38]. The mechanisms the MN system uses to achieve these functions are usually explained by the direct "matching hypothesis" or "motor resonance" in which the encoded neural code of what is observed is matched with a generated neural code of how that movement could be executed [39, 40]. What is being matched could be, high-level abstraction of the intentions, a motor code encoding the plan to emulate a goal-oriented action, or a detailed motor code of the action itself encoding the trajectory of the movement and how to imitate it [41]. Additionally, it has been shown that there exist specific neurons in the MN system that have large specificity towards the way the action is performed and the final goal accomplished, while other neurons lack this level of specificity and the relationship is restricted to the action goal. Other properties of MNs are that they do not activate when observing objects alone, nor when the movement alone is shown [42].

Canonical neurons, on the other hand, seem to encode action possibilities directed towards objects and motivates our incorporation of affordances in a biological model for AU [42–46]. Indeed, action can be understood given both the motion and the goal towards which the action is directed [47].

In this section, we highlighted the need for both biologically inspired processes of environment context understanding and trajectory parsing to be integrated dynamically within a consistent cognitive framework for AU. We decided to model this framework using the Dynamic Field Theory as introduced next.

3 Dynamic field theory

At the core of the modules that make up TARS, CARS and the affordance logic system, are decision making processes that dynamically evolve with the tightly coupled input. These three systems all require cognitive abilities to achieve their functions. CARS requires the cognitive abilities of object detection, motion prediction, and goal selection. TARS, on the other hand, requires feature detection and comparison. Finally, the affordance logic system requires the abilities of dynamic selection and long-term memory. In the following, we present the dynamic cognitive framework of DFT and elaborate on the building blocks that are used within the different systems in this work.

3.1 Dynamics and instabilities

Dynamic field theory (DFT) provides the mathematical and theoretical framework, which builds on dynamic neural fields (DNFs), to model the embodied, situated view of cognition [5]. DNF is a cognitive mathematical model of the dynamic neuronal activation on a population level. It describes decision making inspired by the pattern formation within the cortical neural populations. It is the stable states (localized-bumps) that dynamically evolve (and devolve) in time, given dynamic perceptual input into the neural fields, which provide a unit of representation. These units of representations are a function of the complex interaction between the neurons in the population and are the primary units to describe cognitive properties within the neural fields. The strong recurrent connections between these neurons produce patterns that model detection, selectivity, and working memory. The dynamics are mathematically described in the following integro-differential equation that was initially proposed in [48]

$$\tau\dot{u}(x,t) = -u(x,t)+h+\int f\big(u(x^{'},t)\big)\omega(x-x^{'})dx^{'}+S(x,t) \quad (1)$$

$$\omega(x-x^{'}) = c_{\mathrm{exc}}\exp\left(\frac{(x-x^{'})^2}{2\sigma_{\mathrm{exc}}^2}\right)-c_{\mathrm{inh}}\exp\left(\frac{(x-x^{'})^2}{2\sigma_{\mathrm{inh}}^2}\right) \quad (2)$$

$$f\big(u(x,t)\big) = \frac{1}{1+\exp\big(-\beta u(x,t)\big)} \quad (3)$$

in which the activation of the field $u(x,t)$, as given in (1), describes the activity over the metric dimension x at time t. Here, x, represents a behavioral dimension that the underlying neuronal populations respond to. This behavioral dimension corresponds to a space of features and properties that the neurons encode. Explicitly stated, activity at a certain point in the feature space reflects evidence for that feature value. The amount of activation of the field u can then be understood as the presence or lack of information about a space of features along the behavioral dimension x. The time scale τ describes the relaxation of the field, and the negative constant h defines the resting level of the field. The term $S(x,t)$ describes an external input to the neural field. The integral term conveys the interaction between different field locations. Sufficiently activated field locations contribute to the neural interaction by way of the interaction kernel ω given in (2). That is, the output of the sigmoid function f, given in (3), modulates the activation contribution, given by ω, to other field locations. The sigmoid function with slope β is shown in Fig. 1(a). An example of an interaction kernel ω could be a symmetrical homogeneous interaction kernel with short-range excitation (determined by the amplitude factor c_{exc}, with an area of influence determined by σ_{exc}) and a long-range inhibition (determined by the amplitude factor c_{inh}, with an area of influence determined by σ_{inh}) [49]. Four interaction kernels are shown in Fig. 1(b). The choice of the kernel is usually dependent on the kind of cognitive behavior to be shown. Analysis of (1) leads to the characterization of attractor solutions. In the following, we describe these solutions and their significance [5, 48, 50].

In the case where no external input is present, the field has a constant level of activation, equal to the negative resting level h, along the field dimension. This non-peak attractor state, referred to as a *sub-threshold solution*, maintains its stability under weak external input $S(x,t)$. In the case that the activation level exceeds a threshold level where the lateral interaction $\omega(x-x^{'})$ and the sigmoid function $f(u(x^{'},t))$ become active, the neural field is driven in a different dynamic domain. In this case, a *localized peak* develops in the field due to the increase of activation in the field locations where the external input is the largest [50].

Starting from a *sub-threshold solution*, a *detection instability* can occur in which peaks evolve at positions of sufficient activation. The word instability here is used to indicate a translation between two stable states. The *detection instability* occurs at positions that were successful at accumulating enough activation to overcome the activation threshold of the field. In other words, the saliency of the input, or stimulus strength of that feature-space at that position, or the certainty of the presence of that feature in the current state, was significant. It is possible to have enough activation at several locations within the field and develop localized activity peaks that provide a representation of the existence of the underlying feature-space values. The interaction kernel labeled with number 2 in Fig. 1(b) is an example of a kernel that is used for the detection instability. Furthermore, an example is given in Fig. 1(c) and Fig. 1(d). Figure 1(c) shows an input at a feature position with stimulus strength (solid grey line) that is not sufficient enough to activate the complete field (dashed black line) therefore no information is represented in that field. In Fig. 1(d) the stimulus is strong enough to produce a bump in the field, giving a representation of the existence of information which can be read out for further processing. The interaction kernel used in this example is the second kernel in Fig. 1(b), and that is shown by the fact that the output takes the shape of the kernel around the input's location.

The second case that can be observed is known as the *selection instability*, in which only one stable peak can evolve in the field, and any subsequent activation at different locations in the field is inhibited. Only a large enough activation (one that can accumulate enough activation to overcome the global inhibition induced by the first peak as well as the field's threshold) can appear and inhibit the original peak. When two positions of a quiescent field, that shows the selection instability, show activation at the same time, the one with higher activation develops the peak, and inhibits the other positions, showing a selection of two options. In the case when two or more positions have similar activation values in a field showing the selection instability, noise in the field plays a role in selecting one of the locations to develop a peak. Positions, where peaks of activations are developed, are meaningful as units of representation, and they indicate the existence of an essential underlying value given the selected feature-space. The interaction kernel labeled with number 3 in Fig.

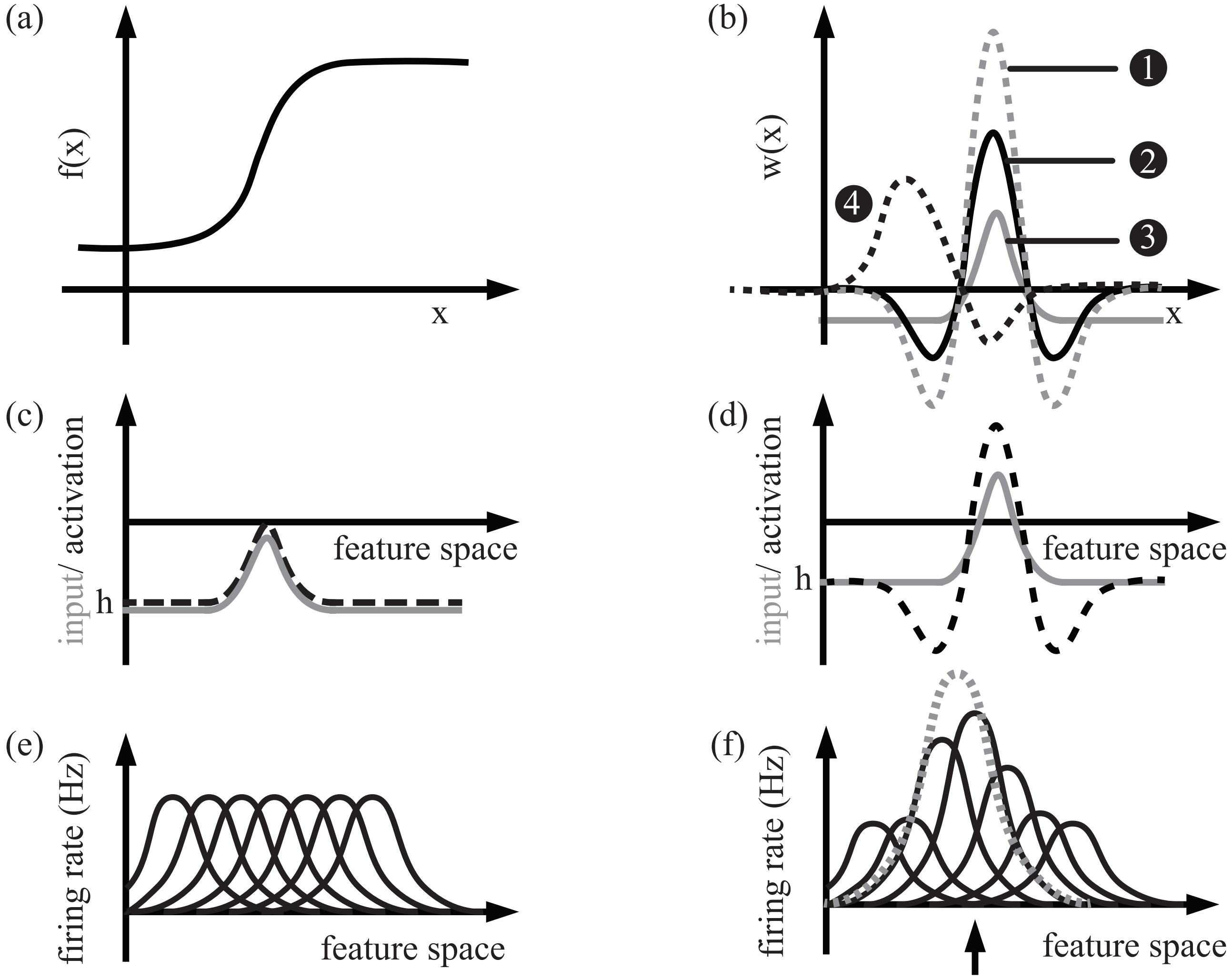

Figure 1. Dynamic neural field components and distribution of population activation. (a) The sigmoid function. (b) Examples of the interaction kernels: 1) An interaction kernel used to model a working memory instability. 2) An interaction kernel used to model the detection instability. 3) An interaction kernel used to model the selection instability. 4) An interaction kernel used to produce a traveling wave transient state. (c) Subactivation solution within the DNF. (d) A field with a stable solution around the input. (e) A group of tuning curves spanning over the features space with no response to a stimulus. (f) The distribution of population activation solution (dashed grey line) to a feature input (indicated at position of the black arrow).

1(b) is an example of a kernel that is used for the selection instability.

An important case that can also be observed in the analysis of (1) is one that models *working memory*. This instability can be observed when sufficient interactions are existent in the field to sustain an input even when these inputs cease to exist. This instability aids in modeling decision/features that were made/observed in the past. The interaction kernel labeled with number 1 in Fig. 1(b) is an example of a kernel that is used for modeling working memory instability. The working memory instability ultimately leads to a self-sustained activation that represents working memory.

In the same way that peaks can be stabilized, they can be destabilized by introducing a negative input to the peak position or by reducing the excitation there. This is referred to as the *reverse detection instability* or *forgetting instability*.

3.2 Dynamic neural fields and distribution of population activity

These elementary forms of cognition (detection, selection and working memory) discussed so far operate on patterns of neural activity representing sensory stimuli or motor control information. To establish this link between neural activity and external stimuli and internal motor actions, the concept of neural tuning is commonly used. The way a DNF can be related to an activity of neural population is through the concept of Distribution of Population Activity (DPA) [51]. An example is given in Fig. 1(e), where 7 (Gaussian approximated) tuning curves span the feature space. The DPA is calculated using the following equation

$$\mathrm{DPA}(x, t) = \left(\sum \mathrm{tuning}_x \times \text{firing rate}(i, t)\right)/N, \qquad (4)$$

where N is the number of neurons whose tuning curves at positions x are multiplied by their activation (firing rate), at time t. The final result of a DPA is shown in Fig. 1(f) where given a feature value, several neurons respond with their firing rate (solid black lines). The final result is visualized with the DPA (dashed grey line). The lateral interaction between those neurons by their activations give way to dynamics within the field as discussed in section 3.1.

3.3 Learning within dynamic field theory

The input that might be used in a field could be processed into a decision, or it could be used to maintain a memory trace over the feature space as a simple form of learning. Learning in DNFs can be understood using what is known as a preshape or a memory trace [5, 50]. It is a formalization that allows retention of stimuli information in long-term memory form. The memory trace, which equation is

$$\begin{aligned}\tau_l \dot{P}(x, t) =& \lambda_{build}\Big(- P(x, t) + f\big(u(x, t)\big)\Big) f\big(u(x, t)\big)\\ & -\lambda_{decay} P(x, t)\Big(1 - f\big(u(x, t)\big)\Big), \end{aligned} \qquad (5)$$

takes input from a DNF with $u(x, t)$, and builds up activation $P(x, t)$ towards the attractor solution (activation-bump) from the input with a time constant τ_l/λ_{build} that is slower than the underlaying DNF. This built up information is lost at a rate that is even slower, τ_l/λ_{decay}, when there is no activation present and models long-term memory. Here, λ_{decay} and λ_{build} are the rates at which the preshape decays or builds up. The constant τ_l is the time constant of learning in the preshape field.

The memory trace is used as a non-activating input to other decision DNFs. It thus acts as a sub-threshold solution to the field, preshaping (biasing) the locations in the DNF and allowing for easier activation if an input at those specific positions are later introduced into the preshaped DNF. Alternatively, a positive homogeneous input to the field (also known as a boost input) would activate those sub-threshold activations in the field.

3.4 Comparisons within dynamic field theory

It is essential to compare different DNFs (e.g., memory trace field and perceptual fields that hold the current input from the environment) to model the recognition of specific, meaningful features in the environment. In addition to the recognition of features in the environment, comparison is essential to obtain a level of satisfaction regarding the completion of an action command that was sent to an intelligent system. To that end, the concept of condition of satisfaction (CoS) was introduced to check if a field had reached a predefined level of activation on one or more feature values [52–54]. In the general case where an intelligent system is a part of the action/perception loop, the action field represents the desired action to be fulfilled. This action field affects the intelligent system by providing set points for the satisfaction of the action. The level of satisfaction is dynamically calculated in the CoS field where the action/preshape field is continuously compared against the perception field. In contrast, in Fig. 2(b), the stimulus in the prescription CoS field matches the learned preshape in the action field, and a decision bump appears in the CoS field, prompting an activation to be detected.

The action and perception fields are an input to a CoS field that indicates if there is a match or not. The CoS field is augmented with a node that gives a logical value of detection or not as shown in Fig. 2(a,b).

For human motion comparison, the CoS suffers from two main drawbacks. Firstly, the CoS field is activated above threshold once a minimum input value of the state of interest is achieved and any further increase in the input is not detected anymore. However, in some comparison tasks, we would like to detect if an input is within a specific range. Secondly, the CoS compares only one specific location in the feature space. However, in our use case, we would like to compare the entire shape of the activation. This would give us confidence that a positive result indicates that the input is of a specific shape as opposed to being activated everywhere. For that purpose we expanded the concept of CoS and propose the concepts of *Range of Satisfaction* (RoS), *Metric of Satisfaction* (MoS) and *Shape of Satisfaction* (SoS).

In this RoS formulation, the action field is used as a pre-activation for both the upper and lower CoS fields. The upper CoS field is also pre-activated with a global negative input with a value that equals the desired range $-R/2$. In the same manner, the lower CoS field is pre-activated with a global positive input with a value that equals the desired range $R/2$. This allows the detection of a feature in the metric space earlier in the lower CoS. Furthermore, it would allow for comparing for a range of activation levels as the upper CoS field would activate and in turn deactivate the RoS neuron. This deactivation aids in checking for the next feature which is an important function when comparing a time-continuous movement such as a reaching motion. An illustration of the function of the RoS is shown in Fig. 2(c).

The MoS concept extends the concept of CoS across the complete metric space rather than just one specific location. The MoS is achieved by negating the preshape input to the CoS field and setting the resting level of the field to zero. This results in activation in the CoS field only when the input exceeds the preshape value. Finally, the similarity of the input to the preshape can be obtained by summing up the activation of the CoS field.

Finally, the *Shape of Satisfaction* (SoS) is the combination of RoS and MoS. Explicitly, the SoS is the RoS however, instead of using the concept of CoS for the upper and lower fields, we use the MoS instead. The SoS allows for the comparison of the shape of the input stimulus with a preshape within a range. Furthermore, the SoS allows for a comparison where the shape of the input is given higher importance rather than it achieving a predefined level of activation.

3.5 Prediction within dynamic field theory

So far, we have discussed several cognitive properties of DFT that can be used as building blocks in any cognitive architecture. We have expanded on the function of CoS to better suite the application of action recognition. However, the *prediction* capabilities within DFT are somewhat limited. Yet, they are vital in an online dynamic application of action understanding. That is why in the following we argue for the need of a mechanism that can look ahead in a feature space and provide predictive capabilities. A transient state that could provide these capabilities can be found in *traveling waves*. Dynamic behavior of traveling activation pulses in the cortical sheets of the brain had been observed [55, 56] and modeled in DNFs [48]. Such dynamics in the neural field has been exploited for intelligent behavior generation [57] and for influencing robotic arm control [58]. Further research on traveling bumps in neural fields have since been conducted and solutions for their collision been modeled [59]. The mathematical formulation of this transient state is given in Appendix D as described in [57]. An example of the kernel required to achieve *traveling waves* is shown in Fig. 1(b) (black dashed line labelled with number 4).

We depend on the different stable states and their instabilities discussed in this section to model *cognitive building blocks* that are used extensively within the CARS, TARS, and the affordance logic as will be discussed thoroughly in Section 5. Firstly, however, we introduce the action recognition task that we had setup to test the developed systems.

4 The action understanding task

For the human action understanding task, we had set up an apartment environment within our laboratory and invited ten participants to perform high-level scenarios as well as short, precise movements we refer to as primitives. The goal of the primitives is to provide our system with learning examples of how simple movements were performed. The concatenation of several simple movement primitives (e.g., walk forward, turn, step forward, reach, grab, pull, etc.) add up to a higher level intention. The primitive actions could be separated into two main categories: manipulation and locomotion actions. The locomotion actions that were recorded were: step (forward, left, right and back), walk (forward and backward), turn (right/left, 90/180 degrees), standing up and sitting down. The manipulation actions that were recorded were: ap-

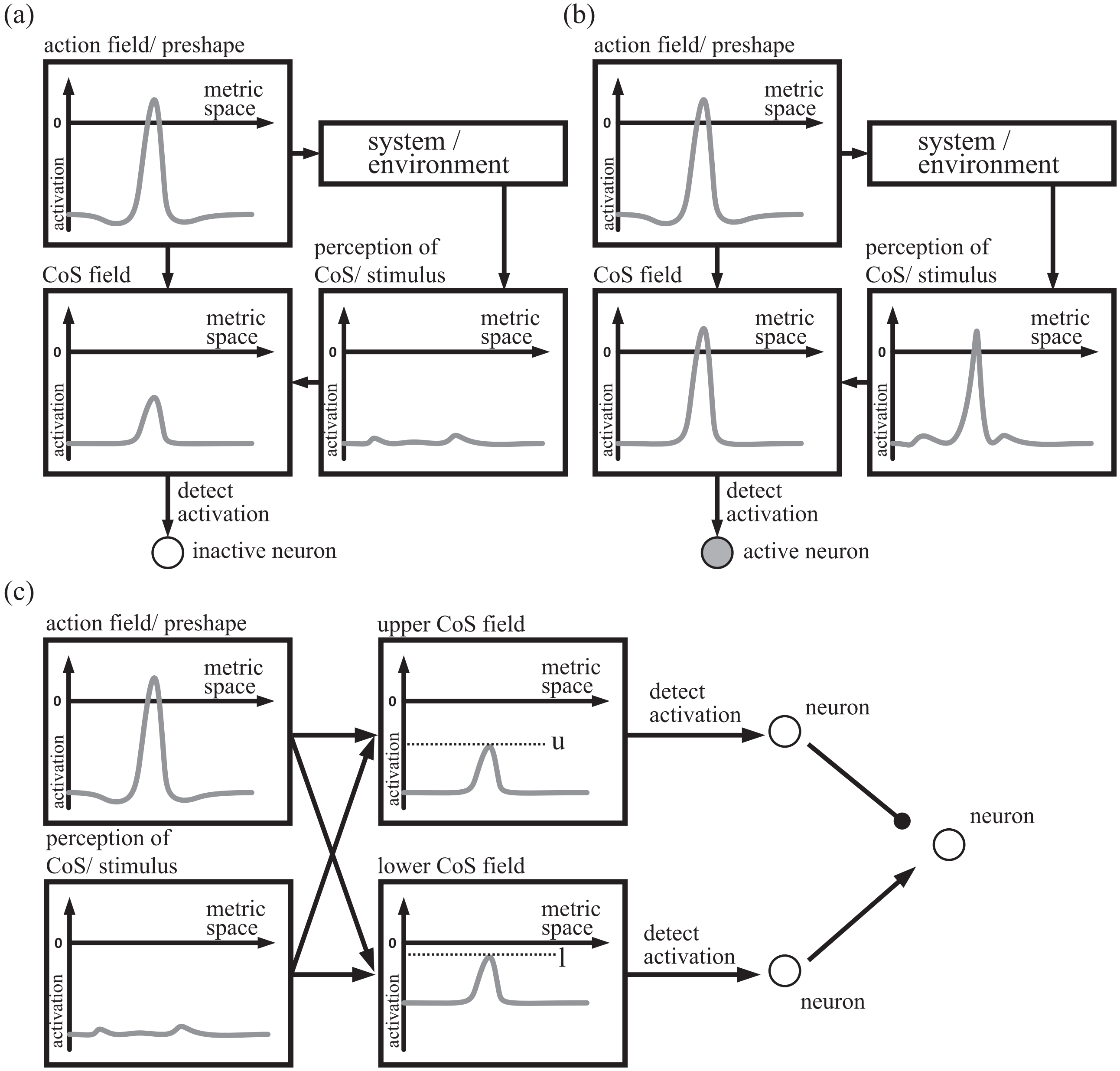

Figure 2. Illustration of the *Condition of Satisfaction* (CoS) approach. (a) Preshaped *CoS field* without corresponding input from the *Perception of CoS*. (b) Matching input resulting in an activation in the *CoS field*, which can be used to activate a neuron. (c) Illustration of the *Range of Satisfaction* (RoS) concept. The preshape and stimulus are used as input for both, the *lower field* and *upper field*. Further, the range boundaries are illustrated within the fields.

proach (reaching action without a grasp, such as turn on the light switch), grasp (a reaching action with a grasp), push, pull, place, open and close door.

We designed the high-level scenarios that portray a specific intention such that a series of primitives mentioned above were used to execute them. The scenarios we set up were: *pick up the remote to watch TV*, *pick up a snack to eat*, *go to work*, *get up on a vacation day* and *tidy up*. The ten participants were instructed to perform the high-level scenarios and were not told to follow a specific order in their execution. We assume that recorded primitives would give us a wide range of movements and allow us to recognize them within the execution of a high-level scenario (intention). By performing the recording session of primitives first, we would prime the participants to using those specific primitives in the high-level scenarios. However, this priming effect was not measured nor analyzed. It was observed, however, that some participants employed creativity and added a lot of character into the high-level scenarios, as one of the instructions they were given was to *act* as if they were in their own apartment. As an example, some chose to do stretching movements in the get-up scenario.

For the motion recording, we used an Xsens MVN full-body inertial motion capture (MoCap) suit. The sensor fusion scheme of the Xsens MVN suit gives the kinematic information (position, velocity, acceleration, orientation, angular velocity and angular acceleration) of each body segment as an output [60]. We opted for a motion capture suit as extracting a skeleton of video frames is not the focus of our work. Furthermore, having MoCap data of movement allows us to model the observing robot anywhere within the apartment environment without being restricted to a specific viewpoint or having to deal with occlusion.

The *grab a snack* task will be evaluated to discuss the results of the CARS in section 6.1. Then, the *pick up remote* scenario will be used to give initial results for the integration of the TARS and CARS in section 6.3. In the following, we will discuss the CARS and TARS individually and then introduce how they can be integrated into an action understanding system.

5 Action understanding system architecture

The systems presented in this work are motivated by findings in situated cognition and neuroscience as discussed in section 2. Explicitly, integration of both systems respects the dynamic coupling between the agent and its environment as the source of intelligent behavior. Additionally, we motivate our approach by descriptions of cognition in which cognition is said to be *enacted* in the sense that cognition arises for adaptive actions [4], and the objects in the environment are represented to reflect their action possibilities and affordances [18, 61, 62]. When observing acting agents in the environment, an observing agent uses its body to understand the observed agent's behavior [63]. Furthermore, the observing agent perceives information directly from the environment and uses the context for understanding and making decisions accordingly. Indeed a major theme in socially-situated cognition is reserved to the idea that the movement and the environmental state of the agents around us are mapped onto the perceiver's body [12]. We expand on the motivation of each of the systems in their respective sections. Concretely put, our hypothesis for modeling the understanding of human action is as follows, the robotic (intelligent) system projects its perspective to that of the acting agent - whose action is to be understood. The robot perceives the affordances directly, relative to the acting agent's body and the environment (objects and their properties). The agent's brain controls the body to localize itself towards objects and to perform manipulation actions. The brain can also observe the own performed actions or of other acting agents. We show an illustration of this workflow in Fig. 3(a).

The abstract blocks and connections, motivated from cognitive studies and neuroscience, illustrated in Fig. 3(a) are translated into the proposed systems and their connections in Fig. 3(b) where the connections between the perception blocks (body and (virtual) objects), the CARS, the affordance logic and the TARS are shown. The suggested future replacement of the preshape block that represents long-term memory and experiences is shown in the hashed motor control/proprioception block.

As discussed in the introduction, the ability to understand the actions of others is a combination of understanding the action possibilities of the goal-directed objects to which manipulations are aimed at, and the spatiotemporal comparison of observed movements to memorized experiences of movement classes. Information from the environment and observed agents are projected onto the observer's body. This processing happens in the *body* block. Our primary hypothesis within this block is that the movements of the actor are seen as the observer's own and the objects around the actor are also projected around the observer [11]. We explain our architecture for extracting neuronally inspired features in section 5.1. When the actor's movement is directed towards an object, the contextual action recognition system (CARS) uses information of op-

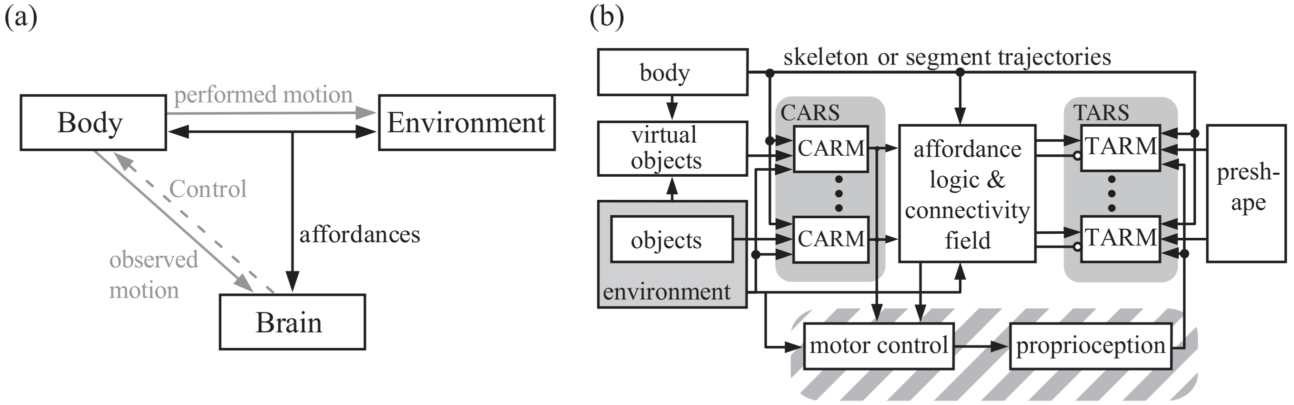

Figure 3. (a) Illustration of the interactions between brain, body and world or environment based on the contextual affordance input as well as the trajectory input information. (b) Connection of contextual- and trajectory based action recognition system. The hatched connections are used to represent possible connections and are not currently modeled in this work.

tical flow (speed and direction of, e.g., the wrists/pelvis) and predicts the object that is to be manipulated. The implementation of the contextual action recognition modules (CARMs) is given in section 5.2. The available affordances of objects (reasoned by an affordance logic block) give an idea of what the meaning of that movement is, this is presented in section 5.4. The trajectory action recognition modules (TARMs), presented in section 5.3, load a memory of a similar movement experienced/learned previously with the help of the preshape block and compare the observed movement to that memory. Each TARM represents a specific action, and thus several of these modules are combined to make the TARS. The internal simulation could also be attained using a dynamic motor control block (shown in the hashed block in Fig. 3(b)), rather than long-term memories currently stored in the preshape block. The use of a human-like motor controller is the subject of our ongoing research and will be integrated with the complete action understanding architecture in future publications. If the action memory is finally validated, then this action is actually being observed, and the system is reset to wait for the next movement. We implemented the architecture using DNFs, where each block in Fig. 3(b) represents the connected neural populations. The next sections discuss the systems that compose this architecture in detail.

5.1 From moving bodies to biologically motivated features

An observer perceives an acting agent as well as the environmental state (in terms of the object in the actor's vicinity and how he interacts with them) to infer about this actor's mental states of actions, (action) plans and intentions. In the following, we present our decisions for modeling the perception of the moving body in a manner consistent with what is given in neurally-focused studies. Specifically, we discuss our choices for how the body is perceived, what are the required transformations, what are the features extracted for the AU task and finally, how these features can be used in a neural population approach that is compatible with the DFT.

5.1.1 Embeddedness and egocentric coordinates

Complying with the embeddedness concept, the observing agent projects the skeleton of the perceived acting agent on his own. Studies have shown that biological motion might be perceived by projection on egocentric coordinates and this might aid guiding behavior and understanding [11, 33, 64, 65]. Similarly, studies in neuroscience and mirror neurons have shown evidence of egocentric action understanding [41, 66]. Therefore, the first step in our action understanding architecture is the projection of the actor's frame of reference onto the observer's frame of reference. Furthermore, the environment the objects in that environment are transformed onto the observer's frame of reference. An illustration of the desired transformation is shown in Fig. 4(a).

5.1.2 The body joint extension and projected relative angle features

Moreover, when observing an acting agent, the observer's visual system focuses on the joints of the acting agent [67]. Out of all the joints, studies have shown that there was a focus on the upper body joints, namely the head, left and right wrists [67]. In our work, we have also integrated the pelvis joint as well as the left and right ankle joints, which are also essential to the understanding of locomotion actions.

The positional information extracted from these joints are then projected onto the transverse and sagittal planes of the observer (after the whole skeleton of the actor had been transformed onto the observer's body frame) [68]. We implemented these transformations mathematically with no regard to possible neural mechanisms behind it. However, transformation-capable DFT systems were also discussed in literature [69] that could also be extended to egocentric coordinate frame transformations for motion perception.

Following from the previous paragraphs, we decided for two *feature types* to be extracted from the projected view for action recognition. The first feature type is the *Body Joint Extension*. It is a non-circular feature (linear feature space, 0-100%) which measures the percentage extension between two joints that are not shared by the same bone. For example the wrist-shoulder body joint extension equals 100% when the arm is fully extended, and 50% when the elbow joint makes a 90-degree angle. We used average human dimensions as given in [70], and calculated the full extension values for a 1.8 meter male for simplification. The second feature type is the *Projected Relative Angle*. It is a feature with a circular feature space (0–360-degrees) which measures the projected relative angle between two joints. Both feature types are described to be view-centered as they are dependent on the position of the viewer relative to the perceived objects (different joints). View-centered representation is one of two major types of descriptions (the other being object-oriented representation) suggested to model the ability to extract information from the projection of a 3D object on retinal images [71–73]. Overall, and given different joints that could be used logically, we propose 39 various features that are calculated for any motion within this work, this accounts for different joints and different plane projections. The full list of features is given in Appendix A. Several combinations of these features can be made depending on the class of the action and the level of joint involvement in that specific movement. Within our work, the temporal evolution of these features is learned from multiple examples to compose a memory that preshapes a comparison dynamic neural field. A memory is learned for each class of action and can be thought as a memorized trace to which the features extracted from the observed action is compared against within the TARM.

The specific choice of the two features mentioned above is motivated by studies of the neural mechanisms behind intentional reaching movements [74–76]. These studies indicate that a reaching motion is decoded from neural populations of directionally tuned cells. Each ensemble of directionally tuned cells is tuned towards a preferred direction of movement. Each ensemble within the population contributes to the population by a vector directed towards the preferred direction of movement specific to ensemble of cells and is weighted by the cells' change in activity. The final sum of the population is called the *neural population vector* and points to a direction close to the observed direction of movement. The intensity of the neural population vector was also shown to be related to the speed or amplitude of the movement. The mirror neuron system suggests that the same mechanisms involved in action generation are the same as those in action perception. Therefore it follows that features for action understanding should be mapped onto the direction and amplitude (distance) of movement [41, 66]. The projected relative angle is a general representation of the direction of movement, while the body joint extension represents the calculation of the amplitude (distance) of the movement. The previous features should be provided as an input to the DFT system in a manner that is neuronally consistent, using formulations within DPA, this process is illustrated in Fig. 5.

5.1.3 Parameter choice for the DPA feature formulation

Tuning curves, centered around the optimal response value, can be modeled using different shapes [51, 77]. For example, they can be Gaussian tuning curves, cosine tuning curves, or sigmoidal tuning curves [78]. The shapes and the parameters of each tuning curve are usually dependent on the specific neuron and stimulus. We highlight the work performed by Perret et al. in [79] and the work of Newsome and Salzman in [80] that investigated the firing patterns in reaching motions, and which we base our work upon. We extracted their results and used the functions they proposed in designing our Gaussian functions

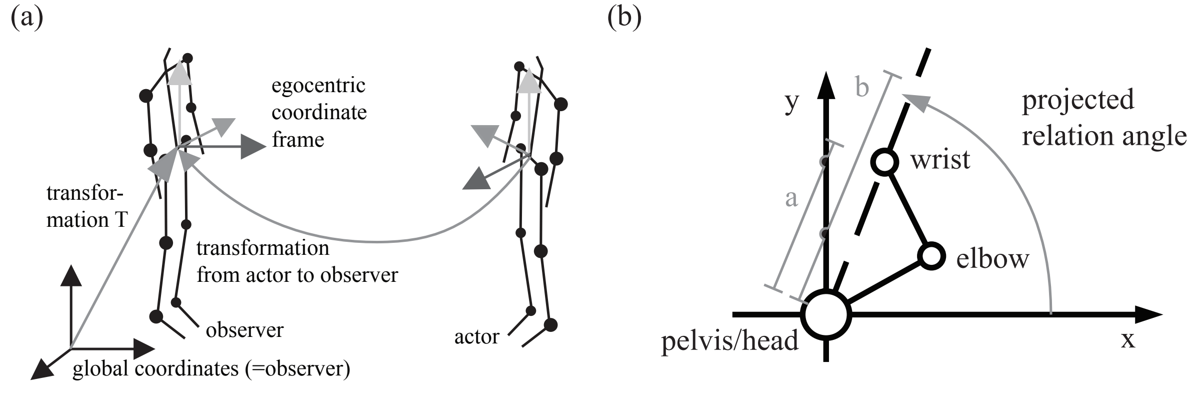

Figure 4. (a) Transformation into egocentric coordinate frame. (b) Illustration of the projected relation angle between wrist and pelvis with respect to the xy-plane. The ratio a/b represents the extension percentage of the arm and is the second used feature.

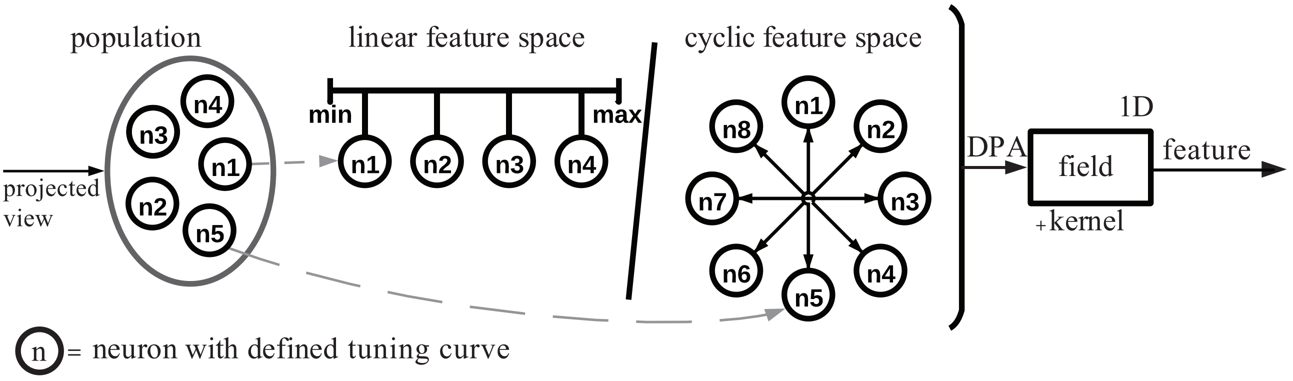

Figure 5. This figure illustrates how features are represented as input stimuli for the overall system (for a linear feature space (left) and cyclic feature space (right)). The tuning curves and optimal response values of the neurons (circles) within the population are defined. Distribution of population activation (DPA) is used to determine the population activation, which is further processed by a DNF to produce the final output.

that represent the tuning curves for motion-sensitive neurons. Further details are given in Appendix B.

For our cyclic features of orientation, we chose eight equidistant neurons representing the feature space. Specifically, the optimal response of neuron is $n_i = f_i, i = 1, 2, \ldots, 8$ where $f = \{0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ\}$. The cyclic features' shape (tuning curves) are modeled after the viewer-centered narrow tuned cell response [79]. For the linear feature space of distance, we used six neurons. The optimal response of each of the neurons was equidistant covering the complete feature space 0 – 100%. The tuning curve of each of the neurons was modeled using a Gaussian function with a wide standard deviation. The Gaussian functions were adjusted using the standard deviation to resemble the results of the fitted tuning curves discussed in the Appendix B and were finally used as they are the standard standards in the DNF framework [50]. The transition from discrete neurons to continuous feature space can be described by the DPA and is used as an input in our work to our DFT architecture. An example is shown in Fig. 6. A stimulus of arm configuration where the projected relative angle was 150° was presented. The dashed grey line in Fig. 6(a) shows the response of the population, while the individual black lines show the individual responses of the individual neurons in the population. While Fig. 6 (a) shows the response for a specific time step, Fig. 6(b) shows the evolution over time in a neural field.

5.1.4 Summary

We have presented our biologically motivated model for motion perception that serves as a pre-processing block for the TARS. The CARS, on the other hand, takes the end effector's/pelvis' direction and speed as an input. The CARM that makes up the CARS is discussed in the next section.

Our choice of features used to encode the 2D traces, shown in Fig. 6(b), was motivated by neuronal optimal response studies [68]. These studies showed that the orientation and distance traveled of observed objects (in our case hand and ankle joints) are encoded neuronally for motion perception [81]. Optical flow, which also encodes a vector of direction and distance of moving interest points, has been shown to be significant of biological motion perception. This is also compliant to what is believed to encode motor commands (preferred population vector for a movement direction), enforcing the notion that the same code that encodes action generation is also used for action recognition [76]. These 2D traces are either saved as long-term memories or provided online for internal comparison with saved memories. The saved long-term memories (preshapes) represent experiences of observing a specific action class [68]. The comparisons are performed in the TARS, however, as the number of actions can be substantial (the number of memories loaded at one time for comparison can be computationally expensive), we provide the CARS which we discuss in detail in the next section.

5.2 The contextual action recognition module

In this section, we propose a contextual system that aids in action understanding. It does so by restraining the search space and obtaining the context of the movement. Our hypothesis in this section is that an intelligent system can extract context from goal-directed motion performed by a human actor by observing the relationship between end-effector (hand) movement and the objects in the near vicinity and their action potentials. In this subsection, we propose an attention-shift model and explain how it was implemented using DNFs.

5.2.1 Motivation and overview

Eye movement has been shown to react to goal-directed movements. Moreover, the relationship between the eye gaze of an observer and the hand of an actor is predictive [82]. Explicitly, in CARS we model the attention shift by the (robotic) observer eyes, from the hands/hip of the actor to the object towards which the movement is directed. The CARS has additional significance since the robotic observer has no option to sense gaze shifts without expensive, invasive gaze detection sensors. Following from the work in [82], and as the gaze of the observing agent follows the actor's end-effector, the chosen feature for the CARM is the optical flow information of actor's end-effector. Optical flow here specifically refers to the direction of motion tracking information. The optical flow information consisting of the actor's end effector (and hip) direction and speed [83] is used as an input for the CARM.

This information is fed to the moving shape module, as shown in Fig. 7, which in turn feeds into a neural field that represents the environment that the actor is performing his actions in.

This moving shape field is initially located at the end effector's starting position and has a specific limit (set by the limit input block) that it is allowed to travel to before it

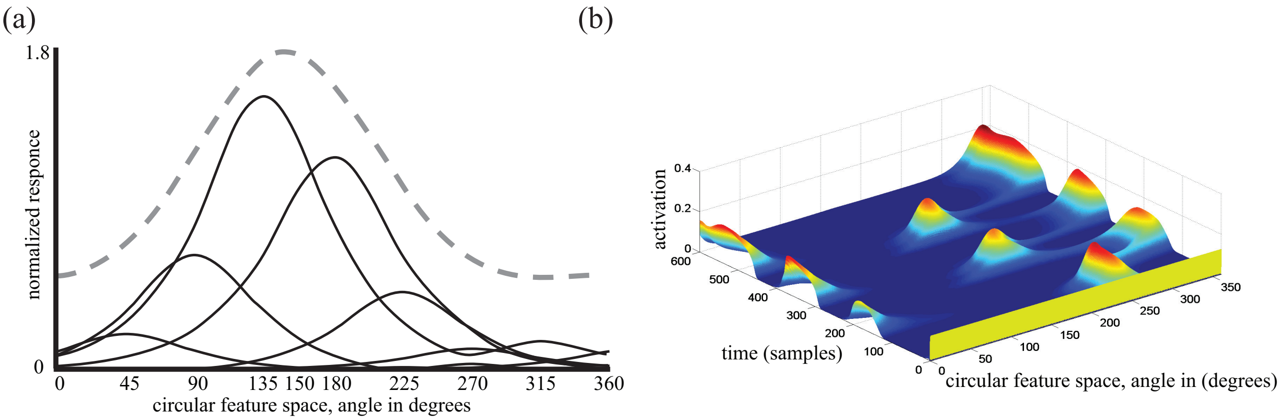

Figure 6. (a) The DPA response for a specific time step given an observed *projected relative angle* of 150 degrees. (b) The 2D memory trace of the *projected relative angle* between pelvis and right foot in the $x - y$-plane for forward walk action.

fades away. The environment neural field is preshaped by the locations (given from the objects block) of manipulable objects in the environment. It is activated if the peaks *shot* from the actor's hand (position is given by the position input block using the direction/speed of the hand calculated using the optical flow input block) hits a preshaped location continuously. The peaks that are shot are calculated in the shape neural field block, the speed of which is controlled by manipulating the parameters in the asymmetric kernel block. The term "preshaped" here refers to the fact that the provided activation of the objects is not sufficient to drive the neural field into activation, the field is then said to be subactivated at those locations or preshaped. In this sense, the environment block does not directly encode the environment per se but the interaction between actor and environment. In the next two paragraphs, we explain the different objects that preshape the environment field and the function of the moving shape module. So far, we have given a high-level overview of the CARM's building blocks. Following, we will give a detailed overview of the object and virtual object input block, the central moving shape module and its inputs, finally the environment field block.

5.2.2 Physical and virtual objects

The object information that is fed into the environment field, as an input, can encode physical objects that preshape the field at the same x, y location they are observed. The same ideas are extended for locomotion actions (e.g., walking, turning, stepping left, stepping right, etc.).Virtual objects are imagined around the actor, and motion directed by the feet or the hands towards those virtual objects would read out their virtual affordances to give a hint of what the possible action is. For example, the stepping forward locomotion action can be understood using the movement direction of the ankle towards a virtual object in front of the feet and so on. While the use of the virtual object is a simplification of how locomotion and free-space movement could be understood, it allows these two classes of movement to be assigned virtual affordances and be integrated into the overall architecture.

5.2.3 The moving shape module

The moving shape module is shown in detail in Fig 7. The inputs to the moving shape module are the *optical flow* input, the *position* input and the *limit* input. The output of the moving shape module is the memory trace activation in the *shape memory* field. The moving shape module contains two fields. The first field is the *shape field* that takes the calculated parameters of the asymmetrical kernel as a first input and the calculated values of the Gaussian means as a second input. The second field is the *shape memory* memory trace that accumulates the output of the *shape field*. Both fields are defined of the metric space field spanning the immediate environment in meters.

The moving shape module models a temporally vanishing memory trace of traveling peaks. The traveling peaks originate from a specific location in the field towards a direction given by the *optical flow* input. The *optical flow* input represents the optical flow of a specific joint e.g. left wrist. The *optical flow* is a two-dimensional input with magnitude and direction terms. This input is used to shape

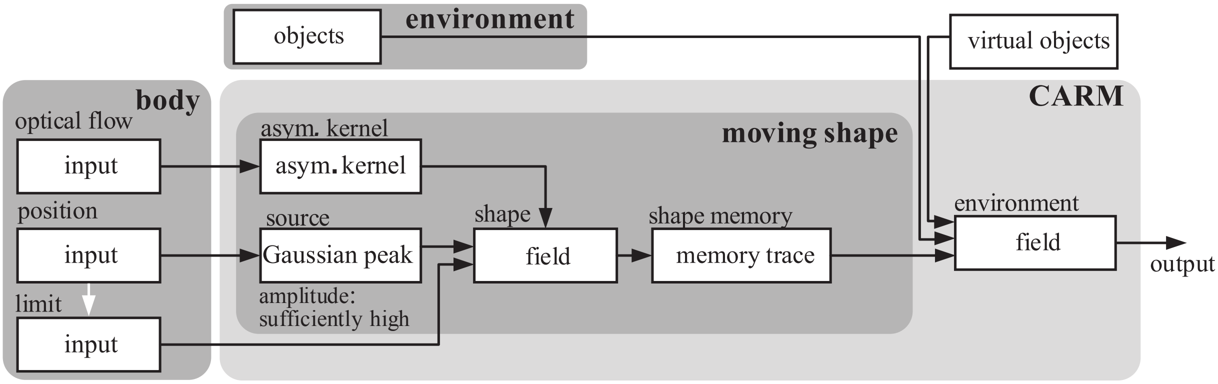

Figure 7. Architecture of the contextual action recognition module

the asymmetrical interaction kernel of the shape field. The *asymmetric kernel* allows the *shape field* to move with respect to the optical flow input.

The shape field in Fig 7 has two inputs: the *source* (which is a 2D Gaussian peak) input and the *limit* input. The position of the 2D Gaussian peak is controlled by the position of a joint (e.g., wrist) $\mathbf{p}(t)$- relative to the shape field dimensions (that is, egocentric coordinates are respected here too) and is always kept at an amplitude sufficient to cause a permanent activation in the shape field. This input is used to represent the position of a specific joint. The combination of the *source* input and the *asymmetric kernel* defines the movement of the Gaussian peak and within the *shape field*. This setup allows for the following activation behavior: an activation peak is *separated* periodically from the *source input* position and travels into the direction of the *optical flow* until it vanishes. The *shape memory trace* saves the activations of the *shape field*. It is important to note that inputs to the shape field are always active. Therefore, there can be multiple moving peaks at the same time. Different moving shapes in the shape field can be created given the optical flow input.

As different activation peaks are *separated* periodically from the *source input* given the *optical flow* input, it was observed that it was hard to control the distance of travel of those peaks as well as their vanishing time. For that reason, the *limit* input was introduced. The *limit* input preshapes (using a 2D Gaussian) the *shape field* to restrict the distance the traveling peaks are allowed to travel. Thus, only regions where the activation peaks are allowed to travel are preshaped sufficiently.

The *optical flow* input is calculated as follows:

$$o(\mathbf{p}(t)) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot (\mathbf{p}(t) - \mathbf{p}(t-1)). \tag{6}$$

The *limit* input is a preshape implemented as a 2D Gaussian function $g(x, y, \mu_x(t), \mu_y(t))$ with maximum amplitude at the current *position* input $\mathbf{p}(t)$ as defined in (7). Accordingly, the expected value μ equals the *position* input $\mathbf{p}(t)$. Depending on the resting level of the *moving shape* field, the Gaussian has to be shifted by c in order to prevent activation within the field (as it should preshape locations where the traveling peaks are allowed to reach):

$$\begin{aligned} &g(x, y, \mu_x(t), \mu_y(t)) = \\ &A \cdot \exp\left(-\left(\frac{(x-\mu_x(t))^2}{2\sigma_x^2} + \frac{(y-\mu_y(t))^2}{2\sigma_y^2}\right)\right) + c. \end{aligned} \tag{7}$$

The calculation of the *asymmetric interaction kernel* $w_{asym}(x, y, \mathbf{o})$ is presented in (8). The basis shape is defined by a 2D Gaussian as described in (7) but without shift c:

$$\begin{aligned} w_{asym}(x, y, \mathbf{o}) =& g(x, y, \mu_x(t), \mu_y(t)) \\ &+ o_x(t)\frac{\partial g(x, y, \mu_x(t), \mu_y(t))}{\partial x} \\ &+ o_y(t)\frac{\partial g(x, y, \mu_x(t), \mu_y(t))}{\partial y}. \end{aligned} \tag{8}$$

A moving shape activation is shown in Fig. 8. This figure illustrates an arm moving towards the right. What this would translate to within the moving shape module are the waves seen in the figure. A moving peak centered at the wrist position would propagate given the information of the optical flow. Accumulating waves would build up activation while noise generated from the movement would die out as shown in Fig. 8. The traces in Fig. 8 can have complex shapes due to two reasons. Firstly, the *moving shape* is dynamically accumulating input as the wrist position changes continuously. Secondly, the memory trace within a CARM maintains the activations in the field, given the field's timescale, thus allowing for complex shapes to appear.

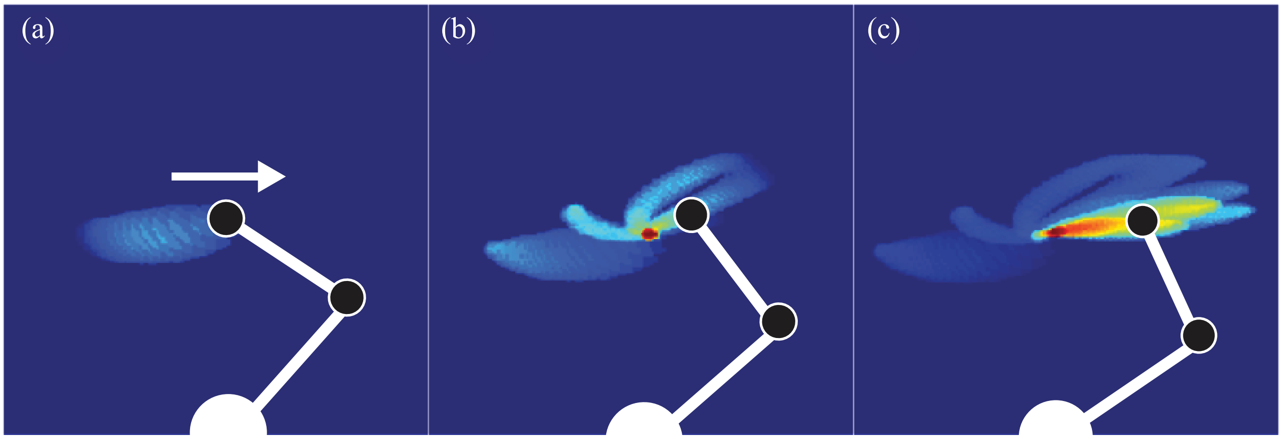

Figure 8. Example of a *moving shape* output. This figure shows three snapshots (order from left to right) of the output produced by a *moving shape* module in the case of non-zero optical flow.

5.2.4 The environment field

Finally, the environment field is a decision field that performs a selection given the (virtual/physical) objects that preshape it and the moving shape module's output that also provides a preshaping input. The field is defined over the feature space representing the environment (in meters). The output is the location of the objects that the observed agent is predicted to manipulate. It is important to note that physical objects in our implementation encode both furniture and manipulable objects. Virtual objects encode positions around the body used for both direction and magnitude (the intensity of the motion) detection.

A stable peak in the environment field is an indication of which object the actor intends to interact with and where this interaction is being (will be) performed. For the virtual objects, it indicates what kind of locomotion movement is being performed and intensity/direction of the movement. The affordances of that specific object can be read out and preshape the TARS which in turn validates the type of affordance on a movement level. We discuss the modules that make up the TARS in the following section.

5.3 The trajectory recognition module

When an acting agent performs an action, his/her movement kinematics provide an abundance of information a human observer could use to recognize the action. In terms of movements, human action varies continuously. That is, for the same action, a person performs movements differently across multiple runs. The time it takes to complete the same action also varies from one trial to another and from one person to another, depending on the task and the kinematics of the actor. In this section, we provide a DNF model of motion trajectory comparison for action recognition that acts independently of environmental information. These different blocks that compose the trajectory recognition module are visualized in Fig. 9. We explain how we achieve spatial and temporal invariance and provide insights on how the intrinsic properties of the DNF could be used to dynamically adapt the fitting between stored memories and the observed data and give it a "better chance" to get a positive fit. We also discuss our implementation for producing and processing these stored memories (templates).

In compliance with the template-matching model, biological systems depend on a stream of features (stimulus) produced by static views of the body to perceive and classify movement patterns [29]. These features can be thought as form cues of a specific body configuration, similar to the concept of snapshots presented in [30]. They are called snapshots of interest within our work. The existence of a specific sequence of snapshots encodes a specific action/movement. We refer to this sequence as the sequence of interest. However, for comparison, we need a reference sequence of interest to be matched against. We rely on a set of stored memories (templates) for the observation of different actions as well as a comparison model. Templates are learned in our DNF model by applying an activation of motion features over time in a DNF that represents a template. The understanding of actions here would be similar to other single-layered exemplar-based sequential approaches that depend on a sequence of feature vectors to perform the classification [6]. We discuss template generation in section 5.3.1. This template has to be adaptive to account for the challenges of AU, for that we present our dynamic template solution in section 5.3.3. From the previous overview, the TARM can be composed into an input side and a preshape side, and they are compared against each other within a *comparison* block, this is discussed in section 5.3.2.

Due to the challenges of AU discussed, the differential speed between the input and the template should be controlled for purposes of correct recognition, and this is done by a *controller block* which is discussed in detail in section 5.3.4. Specifically, the *controller block* controls the speed (and the time intervals) at which the traveling wave propagates through the preshape field, as the stimulus is fed online as the movement is observed.

5.3.1 Template generation

The core mechanism here is the accumulation of a memory trace from multiple samples. The samples, which are represented in feature format, are accumulated within a field with a memory trace. The features, as discussed in section 5.1, encode ego-centric distances and angles between the pose of the head or hip (reference) and the wrists and ankles (end effector) in the sagittal, coronal and transverse planes [64, 67, 68, 84]. The choice of wrists and ankles are because they indeed move the most [85]. The observed agent is projected onto the body frame of the observer such as to achieve view (spatial) invariance and model the internal simulation behind action recognition [33]. The DPA model discussed in section 3.2, was used to model a set of angle and length sensitive neurons at discrete values similar to what is observed in the neural system of the human [75, 83, 86]. The activation of these angle/length sensitive neuronal populations over time activates a DNF either for learning a preshape (template) or to be directly fed as an input for the comparison.

Templates were generated by a mean-like approach within a DNF given several feature examples (the *Body Joint Extension* feature and the *Projected Relative Angle* feature, as discussed in section 5.1) from a class of actions. The template generation process illustrated in Fig. 10 is modeled such that a single observation (in stimulus trajectory form) is appended to the already accumulated motion observations. Our motivation stems from the intuition that an action is observed completely and continuously and is added to overall past experiences dynamically. From multiple examples of an observed action recorded in our dataset, we pick one random sample and present it in stimulus form. This is done in the *select sample* block. The length (time) of the sample is normalized, in the *preprocessing block* to a length that was pre-calculated. This pre-calculated length represents the average length of this specific action class. This input is then fed into two pathways that again merge into a DNF. The upper pathway multiplies the sample with a gain, while the lower pathway accumulates the observations within a memory and multiplies the output with a gain afterwards. These gains are essential to the learning process. They define how the learned information is changed and when to select a new sample to learn from. The two pathways are merged into a DNF that is projected from 2D to 1D such that the time axis is squashed and finally its activation is summed up. A feedback signal (from comparison block to the controller block) is then defined such that this activation summation (which is a proxy of the percentage length of the current example) is compared against a threshold value (which is set to be around 0.95) that determines the transition to learn a new example. The final template is accumulated overtime in the *Memory Trace* field until all examples learnt.

5.3.2 Comparison block

As the learned preshapes could be substantially shorter or longer in time compared to the observed motion, we propose using moving peaks to solve the problem of time variability. A peak would propagate in the DNF of both the preshapes templates and the perceived action. The peak in the preshape would jump to *special* locations characterized by fast changes in the feature space. These jumps would be fast in nature. A jump would occur to the next location in the preshape field only if the same feature was observed in the input field that represents the perceived action. This check is performed in the comparison field as shown in Fig. 9. As the wave in the preshape field propagates more and more towards the end, the more we are sure that the preshape correctly represents the action we think it is.

This *jump* that occurs from one *snapshot of interest* to the other is determined by allowing the wave to propagate forward at high speed and detect areas of interest within the preshape. These areas of interest are either zero-crossing areas or extrema/saddle points. The snapshots of interest are calculated online by merging a Gaussian wave input, and the original preshape as second input in a neural field called the *zero crossing field*. The first input is a Gaussian wave input that is centered around position 0 in the feature space and extended in time. The detection approach using DNFs are shown in Fig.11. The two inputs activate the neural field on intersection within the field;

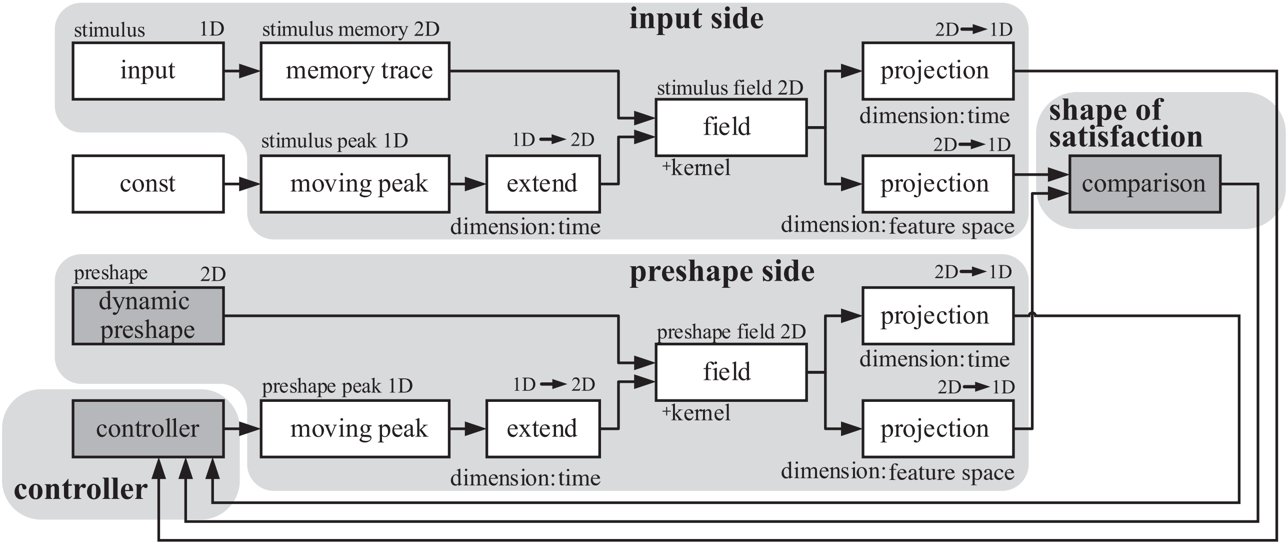

Figure 9. Overview of the trajectory action recognition module. Dark grey blocks represent blocks that are explained in detail in separate sections. Feedback signals are given to the controller from projection fields and the comparison field.

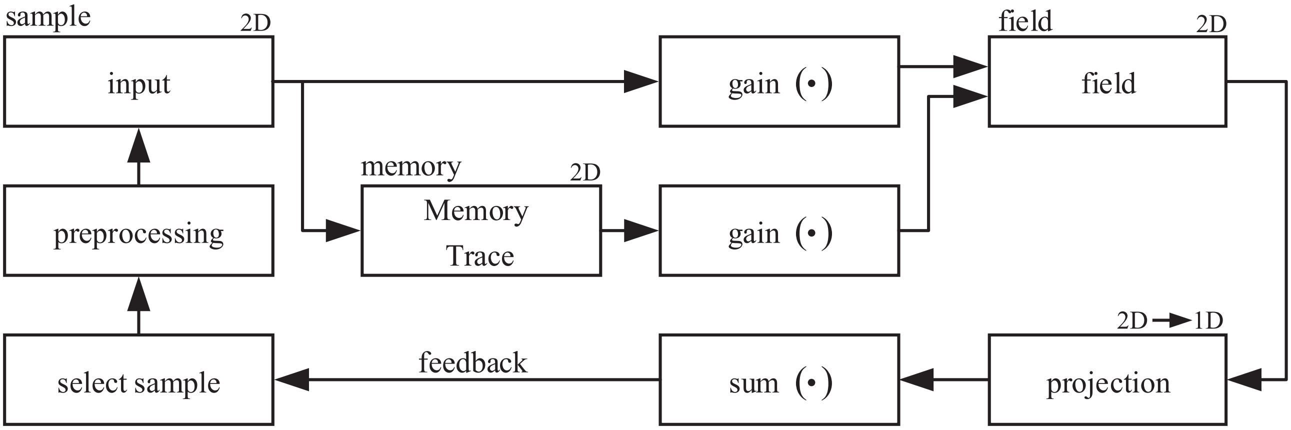

Figure 10. System architecture to generate trajectory-based templates.

this activation is designed to occur around zero crossing points. Special care is given to the calculation of the resting level of the *zero crossing field h*, such that activation occurs when both inputs overlap. The online calculation is of vital importance as the area of interest should be allowed to be shifted and adapted during comparison to allow for the best fit between observed and saved values in the features. The projection of this *zero crossing field* against feature value gives the times at which the sample has zero crossing points. This can be further expanded against time and fed into as an input alongside the original preshape in a field that presents the sequence of interest. The saddle-extrema points can be calculated similar to the zero crossing points, but after an initial derivation of the preshape has been done. The derivation is done offline. The sequence of *Snapshots of Interest* is called a *Sequence of Interest*. An example of a zero crossing *Sequence of Interest* is illustrated in Fig. 12.

Comparison between the snapshots in the preshape and the continuously evolving stimulus input occurs in the comparison block as shown in Fig. 9. The comparison block discussed in section 3.4 is utilized here. The comparison using the concept of SoS allows for the comparison of shapes. Furthermore, the introduction of RoS within SoS, as discussed in section 3.4, provides a level of robustness such that a range of activations can ultimately lead to a successful comparison. The results of the comparison (match/no match or continuous comparison) are used as feedback signals to the controller block.

5.3.3 Dynamic templates

The core mechanism here is dynamically changing the values of the different available parameters (e.g., resting level of the preshape field or the value of the short-range excitation of the interaction kernel) within the *preshape field* given the success of the comparison within the TARM. The underlying motivation behind the set of tools used in the adopted *dynamic templates* approach is twofold. Firstly it is considered a way to allow for a faster successful comparison. Secondly, it is a way to allow the generalization of templates.

As the confidence of observing a specific action increases, the more the dynamic preshape is allowed to influence the action recognition process such as to compensate the spatial variation between the preshape and the stimulus. The portion of the preshape that had not been compared against yet is made to fit the previously observed motion. The compensation is calculated given the past information of the perceived motion. It also allows for the imperfections observed when learning a preshape template and allows some spatial variation between stimulus and preshapes. It aids towards the generalization of the templates. While false positives might be a hindrance due to the use of dynamic templates, the use of CARS would limit the number of loaded preshapes such that this drawback is mitigated as shown in the results section.

The dynamic preshape solution we propose is divided into two steps. The changing preshape step aims at manipulating parameters within the preshape generation method. Such changes could limit the samples used or manipulate the field to exhibit behavior other than producing a mean-like stimulus trajectory. The changing preshape step alters the shape of the preshape entirely and dynamically.

The second adapting preshape step does not change the preshape. It adapts the current preshape given the information seen so far from the stimulus by either shifting it in feature space or influencing its shape slightly. The shape is changed by performing the convolution normally done within the DNF using an adapted 2D Gaussian kernel. The width of the 2D Gaussian kernel is changed depending on the confidence value of the overall trajectory comparison module. This dynamic adaption of the preshape gives a better chance for the fit to occur as we are more confident of our action classification.

5.3.4 Controller block

The controller block shown in Fig. 9 takes three inputs. These inputs are the temporal positions of the moving waves within the stimulus field and the preshape field as well as the results of the comparison block. The output of this block controls the velocity of the moving preshape wave. This controller block is purely an algorithmic implementation and is not implemented using neural fields. Furthermore, we assume for this control block that the length of the input stimulus, and therefore the temporal position of the stimulus within the currently observed action, is not known. This is a logical assumption since we do not know when the actor will end his action nor at which stage he is currently in. We do however assume that we know the length of the preshape and the position of the traveling wave within the preshape. This is again a logical assumption as we have these preshapes stored as memories within our action understanding system.

The controller block controls the velocity of the traveling wave in the preshape field. This is dependent on the feedback signal and is implemented using an approach called *stop and go approach*. The TARS here then is al-

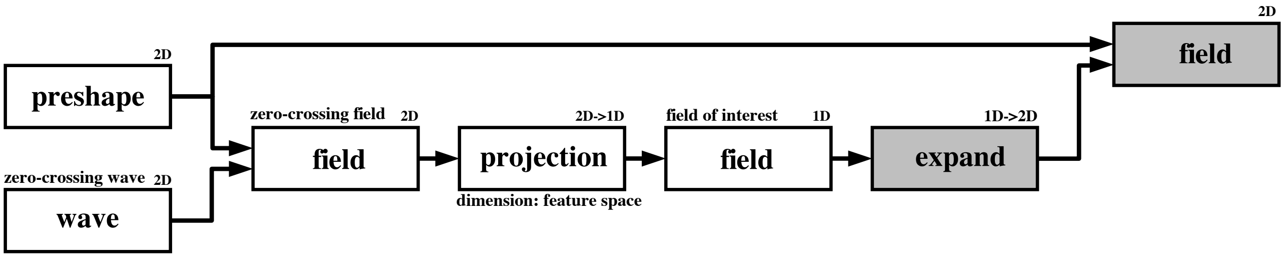

Figure 11. Detection of zero crossing using a normalized preshape and a gaussian wave

ways true for multiple preshapes until the controller cannot compensate the differences between the preshape and the stimulus into adequate velocity commands (too fast or stopping) then it is confirmed that the stimulus does not match the preshape.

The controller, which provides a stop and go signal for the wave, takes a logic input from the comparison module. The controller, which is implemented as an if/else statement, stops the traveling wave (on a snapshot of interest) or allows it to propagate forward with a velocity that is at least as fast as the stimulus' velocity towards the next snapshot of interest. In our implementation the controller sets the velocity of the traveling wave to be twice the velocity of the input stimulus. The overall result of the TARM comparison here can be presented as the percentage of the current position of the wave within the preshape to the total length of the preshape. We define this value as the *confidence* value within this document, which serves as an indication of the correct matching preshape.

5.4 Affordance logic and connectivity fields

The observing agent identifies and predicts the acting agent's action through understanding its dynamic interaction with the (real or virtual) objects in the agent's immediate environment. The CARS shifts the attention of the observer from the end effector towards real or virtual objects whose affordances can be read for further processing. These affordances constrain the set of all possible actions to a limited subset. This subset is used to bias the TARS through a choice of a limited number of preshapes and dismissing the rest. We introduce in the following the concept of *connectivity fields* which aids in achieving the previous ideas.

The connectivity field is a lookup-table-like DNF that encodes future possible object affordances given the object's current affordance state. It houses both ideas of sequential and nested affordances [87]. Each object is represented using its own connectivity fields, which is a 2D DNF with a 2D feature space. The first dimension encodes the current action states of the object and the second dimension contains the action states available in the next time step. As an example, if a glass is being grasped now, it can be released, placed, etc. as shown in Fig. 13. A general structure of connectivity fields is shown in Fig. 13 (a) for a connectivity field of k action possibilities a_{1-k} an object might have. A populated connectivity field is shown in Fig. 13 (b), in which connections were learned in a 2D memory field. The different shades of peaks in Fig. 13 (b) refer to the fact that there exist different probabilities of action transitions encoded in the strength of the connection. Figure 13 (c) shows a learned connectivity field.

Within our implementation, we did not integrate abilities of object recognition nor affordance attribution or learning. Object recognition within DNFs has been discussed in [88]. We assumed knowledge of positions, labels, and affordances of the objects in the environment to be known. Furthermore, the list of affordances was defined in a complementary manner to fit the list of action primitives that were recorded in our dataset. This is following the notion that affordances provide action potentials and provide a logical link between action and environment. These affordances make up the connectivity field.

The connectivity field was realized using a memory trace that saves peaks of activation at connection points between previous and current action state. As the actions are discrete, the input to the memory trace 2D field is an activation of action (neural) population whose tuning curves have no overlap and have an optimal response value spread equidistantly over the feature space. The learning of how current and future affordances are connected occurs as follows: when we observe action changes, both feature spaces activate at the locations of these discrete actions, activation at the intersection of both actions

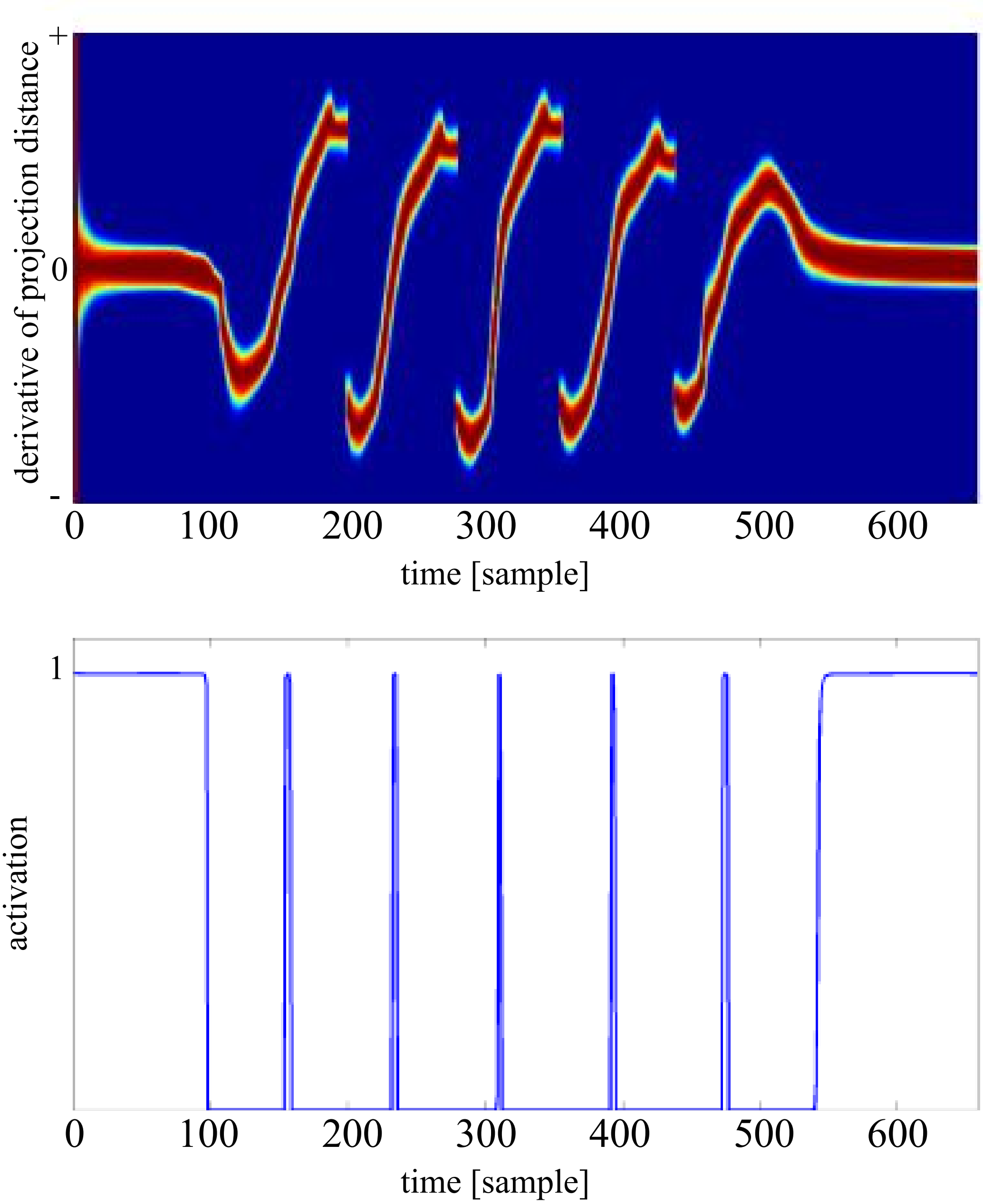

Figure 12. Example for a zero crossing sequence of interest. a) shows the input stimulus of the derivation from a projected distance between right and left foot with respect to the x-y plane over time. b) The corresponding sequence of interest when using zero crossing detection.

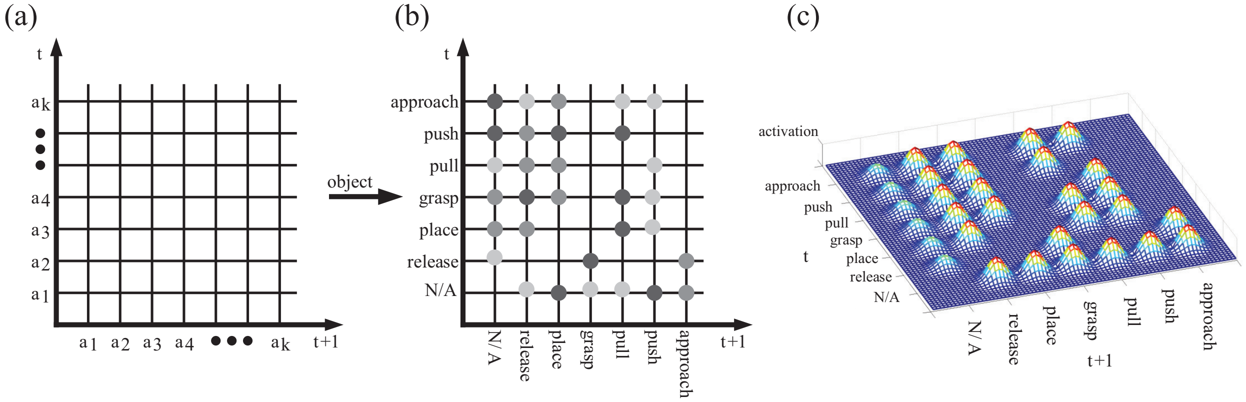

Figure 13. Connectivity field. a) General structure of the connectivity field for k actions a. b) Connectivity field for objects. Gray points represent connections between the current action at t and possible future actions at $t + 1$. Different shades have been chosen in order to represent that different connection strengths are possible.

emerge within the connectivity matrix field. Finally, this peak of activation is saved in the memory trace.

The output of the connectivity matrix can inhibit or excite the saved preshapes of the TARS. When an action is observed it influences the connectivity field. That is, an activation is spread horizontally at the location of that action. This activation is sufficient to activate preshaped peaks (learned in the previous step). These activations are read out by projecting the 2D field onto the next action state axis. These activations go on to excite preshapes in the TARS, and the rest remain inhibited.

6 Results

The previous section focused on presenting the individual modules of the overall architecture. Many TARMs could be recruited depending on the number of actions to be recognized and build the combination of which composes the TARS. Likewise, many CARMs could be used depending on the end effectors, and items that are of interest and the combination composes the CARS. In the following we present our results of the dynamic systems, CARS, and TARS, see section 6.1 and section 6.2. Additionally, we present initial results of the integrated system in section 6.3. The high-level scenario that we used to produce the results in the CARS section is the *Pick up a Snack* scenario. Finally, we evaluate the integrated system with the *Pick remote* scenario. Figure 14 shows the 2D reconstruction of our apartment environment. The *Pick up a Snack* scenario consists of getting up from the couch, walking towards the kitchenette, picking up the apple and walking back to the couch to sit there. The *Pick remote* consists of getting up from the couch, walking towards the TV table, picking up the remote and walking back to the couch to sit there and place the remote on the coffee table. The entire architecture was built using MATLAB/Simulink environment using a modified version of the open source toolbox COSIVINA [89].

6.1 Contextual action recognition system

Three CARMs are running at all times. One for the right wrist, one for the left wrist and one for the pelvis (results for the pelvis are not shown). The virtual objects necessary to be loaded for the function of the pelvis CARM are only loaded when the optical flow information of the pelvis is above a certain magnitude (0.8-millimetres). This threshold was calculated with a decision tree classifier using the magnitude of the optical flow information of the pelvis as the distinctive feature. The moving shape field for the right wrist and the left wrist was shown earlier in this article in Fig. 8, in which a right wrist is simply moving right. The field is preshaped with objects that allow prediction of any interaction.

For our example, the environment houses both furniture items as well as objects. Contextual information of what object and at which location it was manipulated can be inferred using the CARS given the movement of both wrists. We show the results of the CARMs for the right/left wrist interacting with furniture in Fig. 15-left ordinate, and the right/left wrist interacting with objects is also shown in Fig. 15-right ordinate. The right wrist contextual information can be read using the solid/dashed black lines while the grey lines are referring to the contextual information of the left wrist. Figure 15 shows an initial interaction with

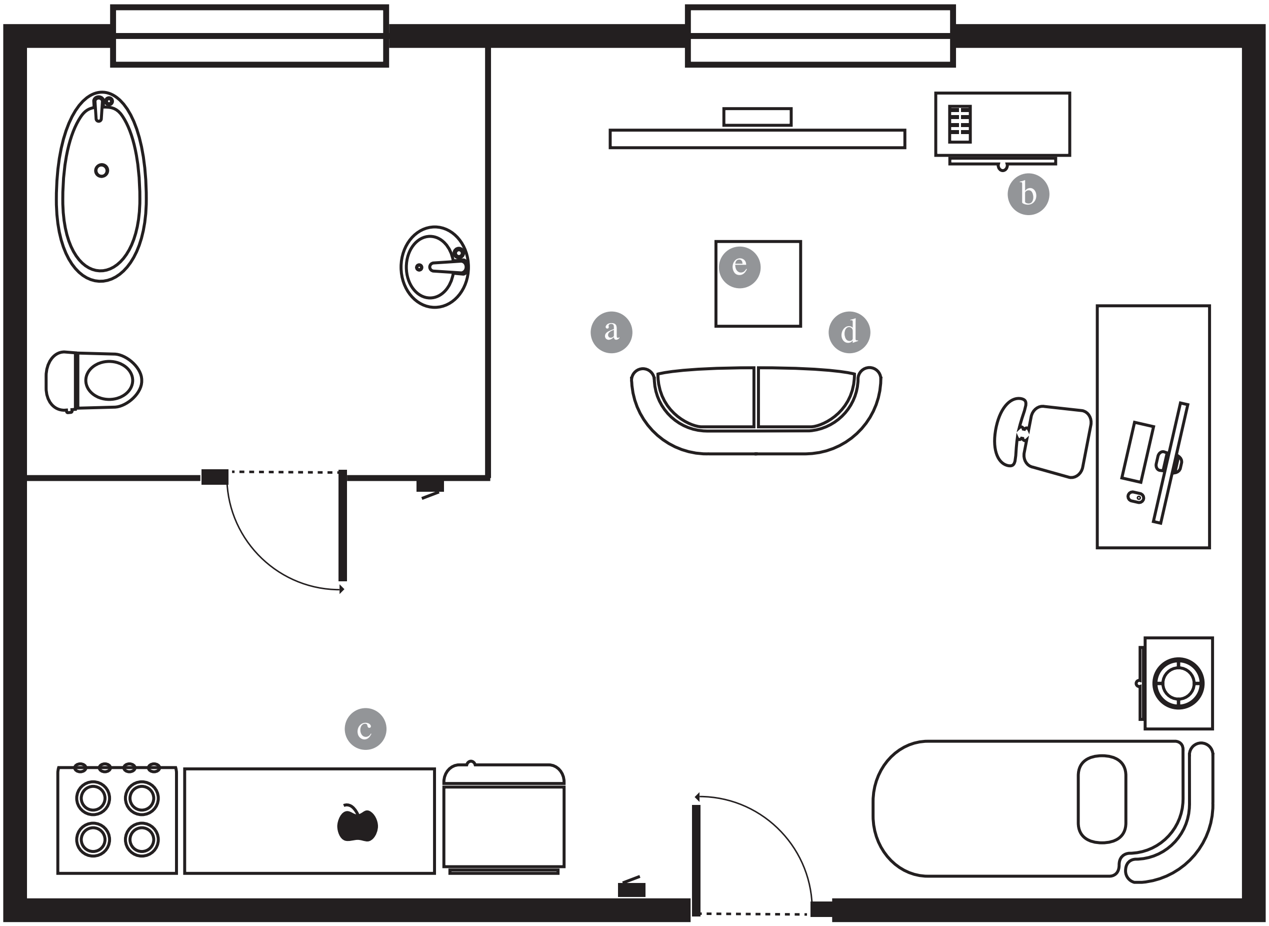

Figure 14. The apartment environment that was used to record the high-level scenarios. (a) The couch (start position). (b) TV table and remote positions. (c) Kitchenette and apple positions. (d) The couch (end position). (e) Coffee table.

the couch (initial sitting position), then as the subjects stands up his/her movement is towards the coffee table and near the apartment walls later he/she interacts with the kitchenette and walks back towards the couch where he/she places the apple on the coffee table. Regarding objects, Fig. 15 shows more extended activation with the apple, as the subject reaches, grabs and walks back to the couch with the apple. As can be seen in the results, the CARS only makes a selection of objects/furniture that are predicted to be manipulated, while suppressing the other objects/furniture. The CARS as expected gives contextual information of *what* and *where* interactions take place. The CARS also gives context of locomotion movements necessary to understand such motion.

6.2 Trajectory action recognition system

Multiple TARMs are running the whole time. One for each action and their respective features. They benefit from the output of the CARS computationally as only a subset of TARMs are excited at each time, the others are inhibited. In the following, we show results for the TARS separately and explain why simple trajectory comparison does not aid in a dynamical action understanding architecture.

Figure 16 shows a comparison of a generated mean for a step forward action, for the feature of projection distance in the $x - z$ plane between right and left foot. Figure 16(a) shows the generated template while Fig. 16(b) shows the corresponding mathematical mean template. Figure 16(c) illustrates the difference of (a) and (b). Thereby, dark red represents the maximum value whereas dark blue represents the minimum. This comparison shows our approach is comparable to the mathematical formulation of a mean template.

This mean template can be adapted dynamically given the results of recognition confidence. The way that the template is adapted within DNF is shown in Fig. 17. Finally, in Fig. 18 we show the results of comparing the step forward action (used as the input to the TARM) against all other action primitives. Only the step variants reach 100% finally, a fault that can be resolved if the CARS was also connected to suppress templates that represent significant movements in the forward direction. However, the confidence level reaches a high level of confidence late, and recognition could be confused earlier across many actions. As these results are obtained by employing TARS alone, the CARS provide means to eliminate a significant portion of these actions and allow for a better comparison as will be discussed in the next section that presents the results of the integrated system.

Table 1: The pick a snack scenario: ground truth

Start (seconds)	End (seconds)	Furniture	Object
0	4.8	couch	
9	10.5	kitchennete	apple
16.5	19	couch	

Table 2: The pick a snack scenario: right hand results

Start (seconds)	End (seconds)	Furniture	Object
0	4	couch	
5.2	6.5	apartment walls	
7.2	9	kitchenette	apple
9	10	bed	
11	13	couch	apple
10.2	19	couch	

6.3 Integration of context and trajectory recognition

In the following section, we show our initial results of the CARS and TARS for the “pick the remote” scenario. Within this example, the participant stands up from the couch, takes a few steps forward towards the television table, picks up the remote, sits back down on the couch and places the remote on the coffee table in front of him. The “pick the remote” scenario’s ground truth is given in Table 3. Figure 19(a) shows the results of the CARS and the objects the observer predicts given the participant’s right wrist movements. The affordances of the objects that are predicted in the CARS step runs several TARMs at the same time as shown in Fig. 19(b). As one TARM reaches a confidence of over 0.8, a decision is made, and an action is then recognized (as shown in the instances marked by the red ovals). The combination of the CARS and TARS then gives a semantic understanding of what are the actions that are being observed. The results of the action understanding system are given in Table 4. In this example the system understands the movements as follows: stand up at couch (0-2.4 seconds) then step forward by the coffee table (2.4-3.6 seconds), turn stepping left towards the TV table and approach and approach the remote (3.6-5.2 seconds), then step forward towards the couch (5.2-6.5 seconds) and finally sitting down on the couch (6.5-9 seconds).

The combination of the two systems alongside the dynamic affordance logic system allows for an end-to-end biologically-inspired architecture for human action understanding. The complete system would benefit from

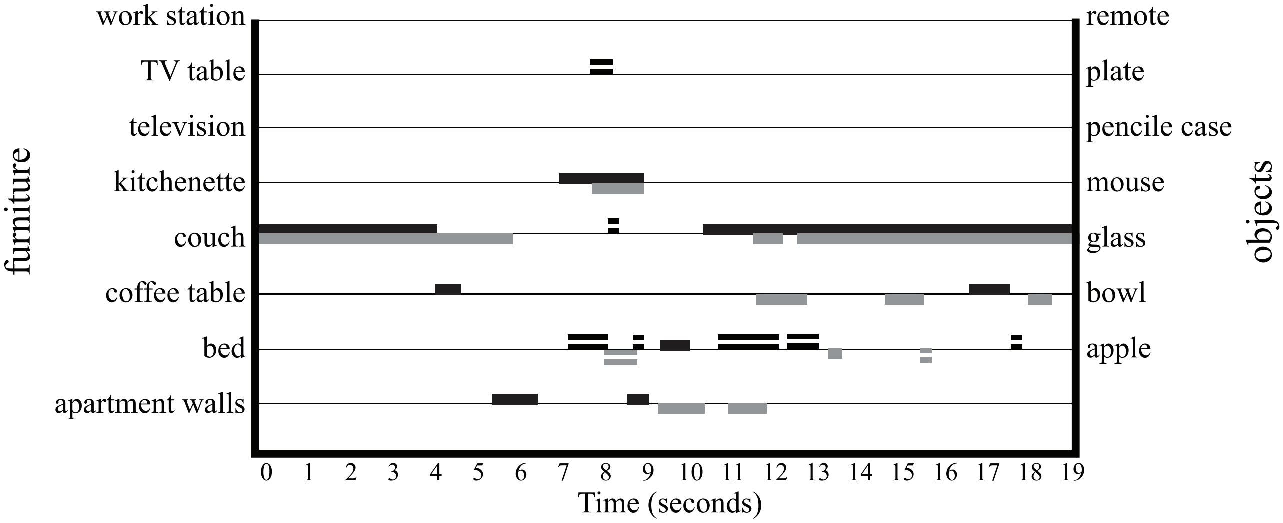

Figure 15. Interaction with the apartment furniture (listed on the left ordinate and read with the solid lines) and apartment objects (listed on the right ordinate and read with the horizontally dashed lines) for the right wrist (solid/dashed black lines) and left wrist (solid/dashed grey lines). The abscissa represents time in seconds. Detected interaction is illustrated by lines. An example of reading the figure would be: the right wrist was interacting with the glass (dashed black line) at the kitchenette (black solid line) around the 8.5 second mark.

Table 3: The pick the remote scenario: ground truth

Start (seconds)	End (seconds)	Action	Furniture	Object
0	0.84	sit	couch	
0.85	2.35	sit-to-stand		
2.36	3.28	step-forward	coffee table	
3.29	4.32	step	coffee table	
3.29	4.32	grasp	TV table	remote
4.33	5.22	step		
5.23	6.09	step forward		
6.1	6.44	turn right 90		
6.45	7.59	stand to sit	couch	
7.6	8.83	sit	couch	

Table 4: The pick the remote scenario: results

Start (seconds)	End (seconds)	Action	Furniture	Object
0	2.4	sit-to-stand	couch	
2.41	3.6	step-forward	coffee table	
3.6	5.2	step left		
3.29	5.2	approach	TV table	remote
4.6	5.3	step left	coffee table	
5.2	6.8	step forward	couch	
6.1	6.44	pull	coffee tabele	remote
6.8	7.5	stand to sit	couch	

an extensive validation given a sizeable human behavior dataset as well as human behavioral studies in intention and action understanding. However, due to space limitations, in this work, we focused on presenting the building blocks (TARS and CARS) and their interconnection. We tested the blocks individually and provided initial results of the integration of these systems to give an insight into the dynamics of decision making. Future work would focus on an extensive validation of the overall architecture. Validation should avoid static representations such as confusion matrices and focus on using new dynamic metrics that measure the conflict between different competing hypotheses of action understanding. Further metrics should measure the interaction between the TARS and CARS modules and weigh the benefit to complexity ratio of combining both signals for a correct and early action understanding. Thus, the proper evaluation of the developed system and definition of metrics constitutes an own research question which will be addressed in our future work.

7 Relation to Related Work

The AU architecture (AUA) presented in this work is a deterministic model that reacts to the input and produces decisions dynamically. This is in contrast to probabilistic

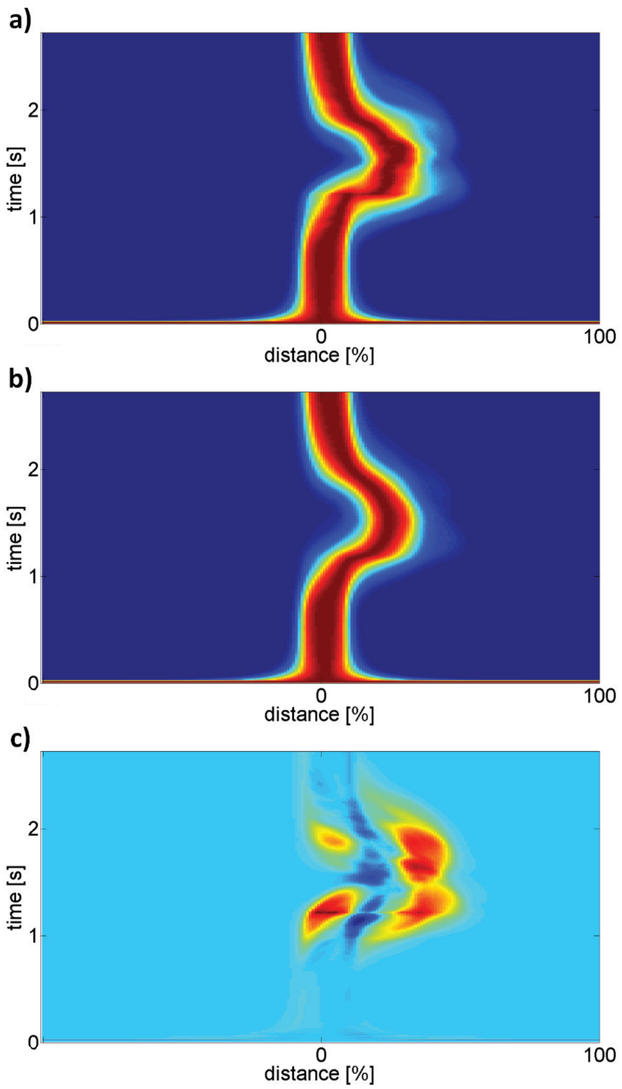

Figure 16. Comparison of a generated mean template with corresponding mathematically calculated equivalent. The chosen example is: “STEP_FORWARD”, projection distance *xz* between right and left foot . a) Generated template. b) Corresponding mathematical mean template. c) Difference of a) and b). Thereby, dark red represents the maximum value whereas dark blue represents the minimum.

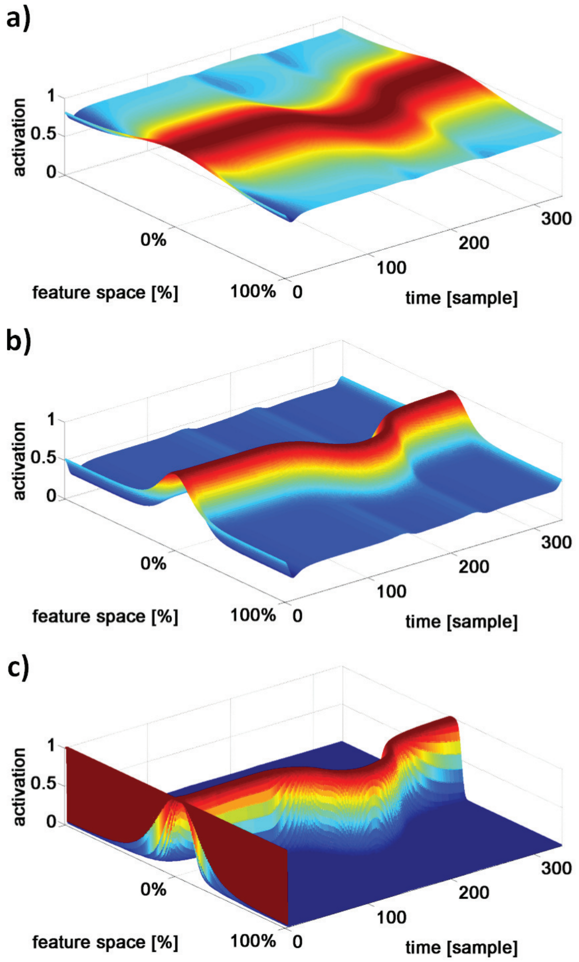

Figure 17. Influence of the adapting kernel for increasing confidence. a) Preshape adapted with a kernel having almost 0% confidence input. b): Confidence is increased to 50%. Finally, c) shows the adapted preshape by 100% confidence, which corresponds the original preshape.

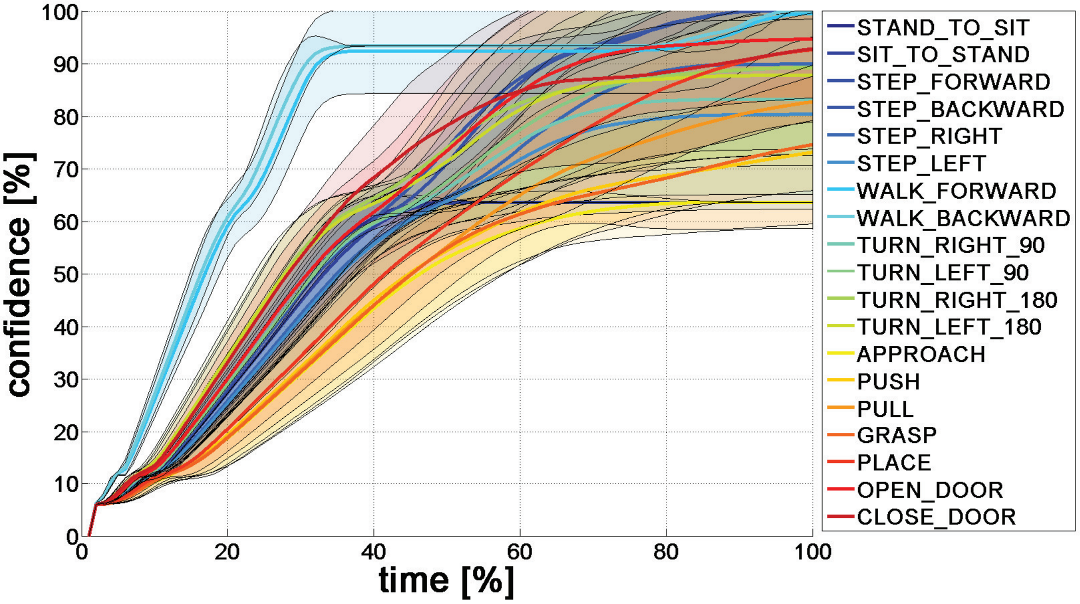

Figure 18. Comparison result of all primitive actions against an input of STEP_FORWARD action. The colored lines represent the mean confidence of the corresponding actions (see legend). The shaded areas around each mean shows the variance. Time has been normalized with respect to the length of the recordings.

models proposed in the literature (we refer the reader to the review [6]). Specifically, we can classify our AUA approach as a dynamic, single-layered exemplar-based sequential method, that depends on contextual information when choosing the example (template). Exemplar-based sequential methods have an advantage of requiring less training data to perform recognition when compared to probabilistic methods [6].

Biologically-inspired AU architectures are usually presented as computational models for Mirror neuron systems. Examples of such computational models are the MOdular Selection And Identification for Control (MOSAIC) model [90–92], and the Hierarchical Attentive Multiple Models of Execution and Recognition (HAMMER) [93, 94] that were primarily developed for imitation and later extended for action recognition [95, 96]. The Mental State Inference (MSI) model [97] as well as the Recurrent Neural Networks with Parametric Bias (RNNPB) [98–100] and the Mirror Neuron System 2 model (MNS2) [101], all model the MNS for AU.

As our model is biologically inspired and resembles the work of Mirror neuron computational models discussed in the previous paragraph, we give a detailed comparison between our model and the previously discussed Mirror neuron computational models in the following. Additional discussion on how CARS and TARS could be related to the Mirror neuron system and other findings in neuroscience is further presented in section 2.4.

Our model resembles the HAMMER architecture in that we do not emphasize a motor control role in the current implementation. This is in contrast to the MOSAIC model that was conceived for purposes of dynamic motor control.

In terms of input, the kinematics of certain joints of interest is used in our model similar to the MSI model. However, unlike other implementations, we explain how features can be represented in population of neurons for action recognition and the generation of long-term memories for each class of actions. In terms of features and logical approach, similar to the MNS and MSI models, we presented a model that gives a central role to the objects in the environment and adopts an object-centered representation. We give this representation further importance and build the CARS to extract information of attention shifts towards objects, select them, read out their affordance and allow this information to bias the TARS. Goal-setting then is a focus in our model, while it is not addressed in MOSAIC, HAMMER and RNNPB models. While other models might allow for goal-setting explicitly, it is not an automatic procedure by any means and the link to the object affordances and motion parsing is not well established, which is what we focus on in our implementation.

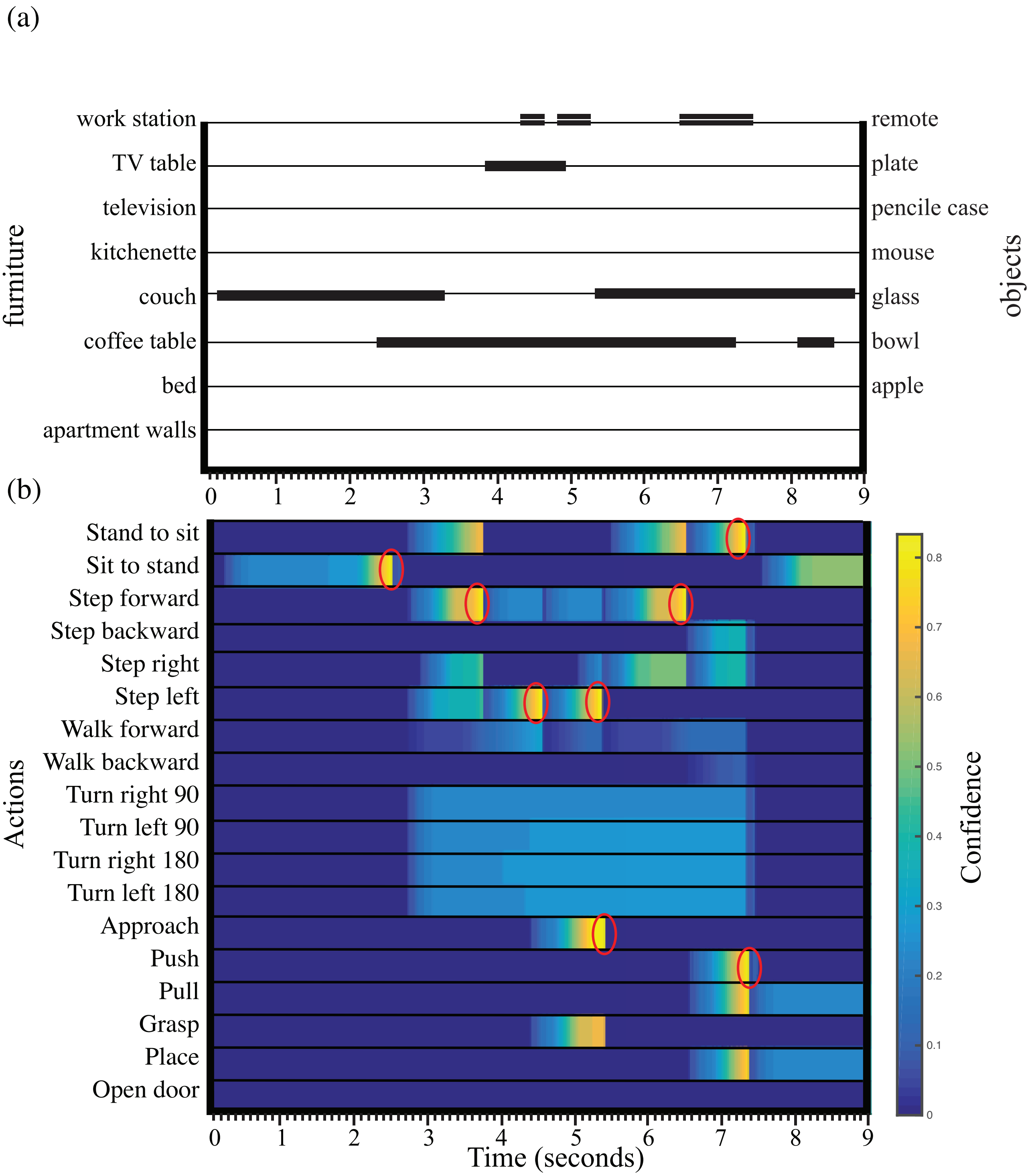

Figure 19. Results for the "pick up remote" scenario (a) Results of the CARS indicating the interaction of the right wrist with the furniture (left, solid lines) and objects (right, solid dashed lines). This indicates that there was interaction with the couch at the beginning and the end of the complete action, with interactions with the coffee table and the TV table in the middle of the complete action. (b) Results of the TARS. Many TARMs are online and comparing the observed movement dynamically, once one of the systems in competition achieves an accuracy of over 0.8, then all systems are reset and wait for CARS to bias the next round of comparisons. The red ovals indicate a decision made.

Projection of the acting agent to the observer is a main block in the TARS which allows the system to be agent-independent and complies with the ideas of "internal simulation" and "motor resonance". This self-observation mechanism is also shared with the MNS and MSI models [102]. However, unlike its use in the feedback-loop for action generation in the MSI model, our implementation uses self-observation in our implementation such as to associate the observed stimulus in an associative memory manner to achieve action understanding. We also address how spatiotemporal variance between the stored long-term memories and the observed data could be handled using dynamic neural fields and to obtain an accurate understanding of the correct motion. Tackling this spatiotemporal variance/similarity between the same/different class of actions has not been addressed in the mirror neuron computational models and is vital for the correct understanding of an action.

All of the discussed models employ a metric to calculate the similarity between the observed or generated (learned representation) of the action. While RNNPB operates on a parameter space, the similarity is calculated based on the distance between the calculated and observed actions. The HAMMER architecture defines similarity based on the completion of the goal. The MSI model, similar to the MOSIAC architecture, simply calculates the instantaneous error (or what is called the responsibility signals in the HAMMER model) based on the difference between the predicted and observed movement. These three architectures, namely HAMMER, MSI, and MOSAIC, in contrast to the RNNPB, operate on trajectory space and thus can calculate the similarity metrics based on the observed/generated motion trajectories [103]. In our model, we obtain an understanding of an action in two steps. First, the CARS selects the object of interest and reads out the possible affordances available at that time step. Secondly, the motion trajectory is parsed in a second step and a decision is made based on the overall activation of a neuron population representing the stored memories of the actions, and how far the traveling wave propagates in that structure.

The setup we proposed within our model allows for online action recognition. Online recognition can also be achieved in the MSI and HAMMER architectures. It can also be achieved in the MOSAIC architecture given the possibility of comparison between different responsibility signals.

Concerning verification, our model evaluates the results on real data of an everyday life scenario. Out of the models reviewed, RNNPB and the HAMMER approach used real data as opposed to simulated data used by the other models.

Other cognitive action understanding systems in the literature that do not explicitly model neuronal processes include the work of Yang et. al in [104], in which context-free grammar and parsing algorithms were proposed for the understanding of goal-directed manipulation actions. The architecture uses a depth image to obtain an articulated model of the user's end-effector as input. The depth image is also used to obtain information about the labels of the objects and their position on a table-top. The hand model is transformed into a set of bio-inspired features which then are used to classify the grasp type using a Naive-Bayes classifier. Additionally, hand tracking produces trajectory profiles for trajectory-based action recognition. The classes were obtained by using a combination of PCA and k-means clustering. An attention model, comparable to our proposed CARS, makes use of bottom-up processes to identify potential fixation points in an image frame as well as top-down attention mechanisms based on the hand location. The spatial intersection of fixation points and the hand location shifts the attention towards an object for monitoring. A new observation consists of a triplet: subject, action, objects. A context-free manipulation action grammar is proposed and using parsing algorithms, a tree group is updated when a new observation is given and dissolved automatically. The tree output can be then passed to an intelligent agent for decision making and further operations.

Other work presented by Aksoy et al. in [105], describes a complex action by combining descriptors that analyze the relationship between the series of manipulated objects with action-related information such as trajectory segments, pose and object information. The combination of these descriptors allows for a better comparison of observed actions and therefore enriches the meaning behind each action. The work describes how observed actions are either understood as new actions or known ones. The new actions are accommodated for by creating a novel schemata, while the known ones, if slightly different are assimilated with the representative schemata.

A neural dynamic approach for parsing a sequence of actions was recently presented in [8] by Lobato et al. The authors present a neural-dynamic architecture that is capable of detecting and representing an even of actions, namely reaching/grasping/dropping objects on a table-top scenario. Trajectory recognition was not considered, but rather three-dimensional positions of hands and objects were used to calculate whether the hand was approaching the object or not. The overall architecture is capable of memorizing a string of actions for overall action understanding.

Neural fields were also utilized for the task of action recognition in the work of Fleischer et al. in [106]. A physiologically inspired model for the recognition of transitive hand actions from video data was proposed. The model makes use of three main components. The first component is a neural shape processing hierarchy that recognizes the moving effector and the goal object. Neural shape processing utilizes Gabor filters, Gaussian radial basis functions and linear regression at the different hierarchy levels to perform recognition tasks of static and dynamics shapes. The first components are also selective for the temporal order of effector shapes to differentiate between, e.g., grasping vs. placing. This is done using the concepts of snapshots within the neural field. The second component models the interrogation of information about the relationship between the effector and the object. A two-dimensional neural activation map named *relative position map* is utilized to understand the relative relationship between the hand and the object such that two features are extracted. The first feature is the position of the hand relative to the goal objects given the assumption of *affordance* neurons that house all possible relative effector positions that constitute a successful grasp. The second feature is the relative motion of the effector in relation to the object which aids in the recognition of e.g., approaching motion vs. moving apart motion. The third and final component consolidates the information from the previous components to model the neural detection of goal-directed actions. The work presented by Fleischer et al., in contrast to the work presented here, uses the only the relative movement information to recognize action and does not take the shape of the trajectory into account. Furthermore, the model takes as an input a sequence of video images as to detect the effector and object locations.

Overall, the AU architecture in this work presents a novel predictive system within DFT, models attention-shifts, and pairs up with a trajectory parsing system in a second step. The trajectory parsing system takes account of spatial as well as temporal variations that are usually problematic when understanding actions. Particular attention is given on how objects and the environment are integrated into the overall architecture and on how they can drive action understanding.

8 Discussion

There is an infinite set of intentional descriptions consistent with any given behavior stream. However, even though there exists a significant state space of possible interpretations, adults seem to be skilled at agreeing about the semantics of an observed action to a detailed description [107, 108]. Even from a young age, we can understand actions (e.g., grasping, pointing and gazing) and attribute a meaning behind them accordingly [109, 110]. These social abilities of action, plan, and intention understanding that we possess as humans allow us to interact with others around us socially.

We presented an action understanding architecture that aims at understanding human actions through the integration of movement and context in a dynamic framework that links bodies, brain, and environment in an embedded cognitive fashion. We introduced an attention shift model that has an application in the CARS and a trajectory comparison model that has applications in TARS. We also introduced how the link between CARS and TARS could be logically motivated using the concept of affordances and connectivity fields.

A biologically motivated approach for feature selection and generation was discussed. While the features in this work were calculated for a generic 1.8 m tall male, given the actual height and weight of the observed actor, the whole anthropometric measures (and thus the features) can be derived using correlation formulas [111]. The features calculated, encode relations between different joints in the body. It would be beneficial to devise a system that dynamically switches between different feature sets for enhanced recognition and reduced computational load. Considering features that encode end-effector–objects relations could also be in line with the current work and would enhance recognition rates. Overall the implemented 39 features were sufficient to test the current system and produce the results as seen in the results section.

The features themselves were represented and fed into the DFT architecture using a biologically motivated approach, namely the DPA method which integrates naturally with the concept of dynamic neural fields [112]. We focus on representing the tuning curves in a way that is consistent with neural response studies in the literature. The assumption that tuning curves are the same across the population is a limiting one. Indeed, it might be the case that the shape of the tuning curve can be different. Moreover, our assumption of equally distributed tuning curves across the feature space is simplistic and maybe not biologically plausible. We assume that the tuning curves are the same across the population as well as being equally distributed over the feature space as a simplification. Further, work on how and what it means for the optimal response values (both in value and quantity) to be optimally distributed along the feature space might allow for a more meaningful stimulus generation for the DFT architecture.

An attention-shift model was developed for context understanding in action recognition tasks. The bias introduced by the CARS aims to reduce the overall computational complexity of the system. The idea that an observer's expectation of a movement effects how the intention behind it is understood has been shown previously in literature [113]. Furthermore, the need for a top-down mechanism to constrain intentions of an actor has been discussed in [114] where the Gricean pragmatic analysis of language (specifically the reality and cooperative principles) were used as the constraint to the understanding of simple, goal-oriented actions.

Regarding neural plausibility, the online computation of the optical flow is problematic, however. This is because the online calculation of the optical flow would require rapid and precise plasticity in the synapses that implement lateral interactions. As such, the implementation of CARS should be seen as an algorithmic shortcut for a more complex neural system that could generate moving peaks as described in the CARS implementation.

The TARS subsequently load only a few preshapes that are dependent on the input from CARS. Furthermore, as an internal comparison is the basis of the TARS, the current implementation depends on a learned memory of how the movement evolves. The template generation methods produce preshapes that are useful for the comparison process. However, two major issues with the production of template preshapes had been observed and been tackled, namely the branching and widening effects. Branching occurs when there are multiple ways of performing an action kinematically (in contrast to having one way with small variances in motion). In this occurrence, we can observe a branch in the preshape that starts from a common point and ends separately. The branching effect has been solved by post-processing these preshapes into a DNF that ultimately picks between branches (the one with most activation) and eliminates the other. The other issue is widening, which refers to the fact that the preshape can take a wide range of features at some parts due to significant variations in the performance of an action. These wide areas usually survive in the post-processing procedures and could facilitate erroneous detection. This has many limitations; specifically, an action recognition system can not house all possibilities for the same action (different speeds/ extensions) that could encode the same action class. We have tackled this problem by trying to adapt the preshape dynamically as well as using a temporally invariant comparison method (traveling waves and extracting snapshots within the learned memory/ preshape).

The CARS and the TARS are brought together such as to limit the search space using ideas of affordances embedded in the connectivity matrix. Using affordances, however, is not without complexity. Further, work should focus on how objects' action potentials are perceived and modeled within DNFs, similar to that presented in [115]. DNF can be used to attribute affordances to objects by modeling a neuronal pool of properties that should all be present for a given affordance to be turned on. Additionally, affordances could be attributed to objects over time by observing goal-directed movements towards new objects, recognizing the actions dynamically and associating the complementary affordance to that object. Affordances suffer from a more elementary problem of definition. It is not clear how it is fully defined, and its implementation should model the state of the agent's brain, body, and environment as we have presented in this work. That being said, the current system architecture cannot learn new affordances, or detect new objects in the environment. For a more meaningful and correct implementation of a cognitive architecture, these issues should be addressed. Furthermore, detecting and learning new action abilities (primitives) is also not implemented. However, if affordances could be attributed in future work, new actions that trigger these affordances could be learned online given affordance understanding.

Given the above discussion, we describe in the following how the different modules interact with each other using the "pick up remote" scenario presented in the results section. Initially, as the observed agent moves around in the environment, its skeleton is transformed onto the observer's egocentric coordinate frame as discussed in section 5.1.1. Furthermore, the *Body Joint Extension* and the *Projected Relative Angle* features (list of features given in Appendix A) are calculated as discussed in section 5.1.2. As the pelvis and wrists of the agents move, they provide input to their CARMs to detect a manipulation movement (towards an object/ furniture) or a locomotion movement (towards a virtual object). In our example, the agent is interacting with the *couch*. The CARS decides that the couch is being *manipulated*. This affects the affordance logic block to activate the TARMs that are related to the couch, e.g., sit-to-stand action or stand-to-sit. The TARS loads the appropriate TARMs (sit-to-stand action and stand-to-sit), allowing the prehshapes of each action across the different features to be loaded. The comparison occurs in each of the TARMs relating to the action/feature pairs against the observed motion as discussed in the comparison block in section 5.3.2. A decision is achieved within the TARS as one of the TARS achieves an accuracy of 0.8 or above. This resets the system and waits for the CARS for the next input. In this case, it is recognized as a locomotion action (the attention shift was towards a virtual object), which forces several locomotion TARMs to turn on as well as the new affordance of the coach (sittability, now that the couch is available to be sat on again). The next round of TARS detects that the agent is stepping forward, and so on until the end of the complete series of intentional actions.

Compared to similar work in literature, we presented a novel predictive system within DFT models attention-shifts and pairs up with a trajectory parsing system in a second step. Special focus has been given to how kinematic trajectories are introduced into DFTs and how the comparison could be performed regardless of possible spatiotemporal variations between the completed and saved representations of the actions.

9 Conclusions

This article presented, for the first time, two systems that are hypothesized to be central to the task of action understanding. The two systems were realized within DNFs. The first of which, TARS, takes information of movement kinematics. The CARS, on the other hand, takes information of movement kinematics, object locations as well as affordances in the environment. The two systems produced cognitive decisions that answer questions of what is the action that is being performed, where it is being performed and towards which object, all within the theory of dynamic fields. The success of the two systems stems from the tight, dynamic coupling between the environment and the decision-making units. This allowed for the production of contextual information necessary for further processing. The initial test results generated using the integration of the two systems provide an essential step towards a robotic cognitive ability of mental state estimation and intention understanding. Further work should focus on further validating the complete system using the recorded dataset to evaluate the accuracy of the architecture. In future work we also aim to extend the realized system with action production modules to augment the long-term memory templates, that are currently being used within TARS, to achieve internal simulation of predicted actions. Human-centered studies could also be designed to validate the need for a contextual system alongside a trajectory recognition system to understand a sequence of actions.

Acknowledgement: This work is supported in part by the MOBOT project within the 7th Framework Programme of the European Union, under the grant agreement n.600796 and the Institute of Advanced Studies of the Technische Universität München.

Appendix A: List of calculated features

The list contains all calculated features, providing the feature name, body plane (coordinate frames given in figure (4)) and corresponding Xsens joint and segment names:

1. projected relative angle XY Head: head angle projected on the XY plane
2. projected relative angle XY LeftFoot: left foot angle projected on the XY plane
3. projected relative angle XY LeftWrist: left Wrist angle projected on the XY plane
4. projected relative angle XY RightFoot: right foot angle projected on the XY plane
5. projected relative angle XY RightWrist: right Wrist angle projected on the XY plane
6. projected relative angle XZ Head: head angle projected on the XZ plane
7. projected relative angle XZ LeftFoot: left foot angle projected on the XZ plane
8. projected relative angle XZ LeftWrist: left Wrist angle projected on the XZ plane
9. projected relative angle XZ RightFoot: right foot angle projected on the XZ plane
10. projected relative angle XZ RightWrist: right Wrist angle projected on the XZ plane
11. projected relative angle YZ Head: head angle projected on the YZ plane
12. projected relative angle YZ LeftFoot: left foot angle projected on the YZ plane
13. projected relative angle YZ LeftWrist: left Wrist angle projected on the YZ plane
14. projected relative angle YZ RightFoot: right foot angle projected on the YZ plane
15. projected relative angle YZ RightWrist: right Wrist angle projected on the YZ plane
16. body joint extension XY Head LeftWrist: percentage of extension length between the head and left Wrist in the XY plane.

17. body joint extension XY Head RightWrist: percentage of extension length between the head and right Wrist in the XY plane.
18. body joint extension XY LeftFoot RightFoot: percentage of extension length between the left foot and right foot in the XY plane.
19. body joint extension XY Pelvis Head: percentage of extension length between the pelvis and head in the XY plane.
20. body joint extension XY Pelvis LeftFoot: percentage of extension length between the pelvis and left foot in the XY plane.
21. body joint extension XY Pelvis LeftWrist: percentage of extension length between the pelvis and left foot in the XY plane.
22. body joint extension XY Pelvis RightFoot: percentage of extension length between the pelvis and right foot in the XY plane.
23. body joint extension XY Pelvis RightWrist: percentage of extension length between the pelvis and right Wrist in the XY plane.
24. body joint extension XZ Head LeftWrist: percentage of extension length between the head and left Wrist in the XZ plane.
25. body joint extension XZ Head RightWrist: percentage of extension length between the head and right Wrist in the XZ plane.
26. body joint extension XZ LeftFoot RightFoot: percentage of extension length between the left foot and right foot in the XZ plane.
27. body joint extension XZ Pelvis Head: percentage of extension length between the pelvis and head in the XZ plane.
28. body joint extension XZ Pelvis LeftFoot: percentage of extension length between the pelvis and left foot in the XZ plane.
29. body joint extension XZ Pelvis LeftWrist: percentage of extension length between the pelvis and left Wrist in the XZ plane.
30. body joint extension XZ Pelvis RightFoot: percentage of extension length between the pelvis and right foot in the XZ plane.
31. body joint extension XZ Pelvis RightWrist: percentage of extension length between the pelvis and right Wrist in the XZ plane.
32. body joint extension YZ Head LeftWrist: percentage of extension length between the head and left Wrist in the YZ plane.
33. body joint extension YZ Head RightWrist: percentage of extension length between the head and right Wrist in the YZ plane.
34. body joint extension YZ LeftFoot RightFoot: percentage of extension length between the left foot and right foot in the YZ plane.
35. body joint extension YZ Pelvis Head: percentage of extension length between the pelvis and head in the YZ plane.
36. body joint extension YZ Pelvis LeftFoot: percentage of extension length between the pelvis and left foot in the YZ plane.
37. body joint extension YZ Pelvis LeftWrist: percentage of extension length between the pelvis and left foot in the YZ plane.
38. body joint extension YZ Pelvis RightFoot: percentage of extension length between the pelvis and right foot in the YZ plane.
39. body joint extension YZ Pelvis RightWrist: percentage of extension length between the pelvis and right Wrist in the YZ plane.

Appendix B: Shapes of the tuning curves

In our search for the most representative shapes of tuning curves, we investigated the results of the work of Newsome and Salzman [80] and the work of Perret et. al. [79]. The work of Newsome and Salzman focused on the direction discrimination in monkeys. They measured the visual response from the direction column in the middle temporal visual area (MT). We investigated their recorded data that presented the intensity of response given the direction of motion of the shown stimuli. After initially testing with cubic spline fitting, and parameter minimization using different family of curves, we settled on a representation using a Mexican hat function $\psi(x, \sigma, c)$, with width (standard deviation) of σ and offset c. The parameters of which were decided by solving an argument minimisation problem (equation (9)) that minimized the Euclidean distance between the fitted spline $s(x)$ and the Mexican hat $\psi(x, \sigma, c)$ given in (10).

$$\underset{\sigma, c \in R}{\arg\min} \sum_{x \in [-180, 180]} |s(x) - \psi(x, \sigma, c)| \tag{9}$$

$$\psi(x, \sigma, c) = \frac{2}{\sqrt{3\sigma}\pi^{\frac{1}{4}}} \left(1 - \frac{x^2}{\sigma^2}\right) \exp\left(\frac{-x^2}{2\sigma^2}\right) + c \tag{10}$$

The work in Perrett et. al. also provides measured tuning curves and analyzed them. We investigated results in their work in which they record neuronal responses to different head orientations and used their data in our modeling. The results showed that body parts are represented using view-centered descriptions. Furthermore, cells can be described as broadly, bimodally or narrowly tuned. We used the cell response information to model the tuning curves using a modified version of the fitting function (equation (11)) used in their original work. Perrett et al. argue for their choice of this equation stating that "it makes few assumptions about the nature of view tuning" [79]. Our modified version (15) guarantees symmetrical tuning curves and was used to solve the optimization problem in (12). Firstly however, the parameters β_{1-5} that compose (11) have to be approximated given the extracted data $d(x)$ using the minimization (12). Therefore modifying (11) to fulfill the condition $R(x) = R(-x)$ we get (15) following these steps:

$$R(\theta) = \beta_1 + \beta_2 \cos(\theta) + \beta_3 \sin(\theta) + \beta_4 \cos(2\theta) + \beta_5 \sin(2\theta) \tag{11}$$

$$\underset{\beta_{1-5}\in R}{\arg\min} \sum_{x\in[-180,180]} |d(x) - R(x, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)| \tag{12}$$

$$\begin{aligned} &\beta_1 + \beta_2 \cos(\theta) + \beta_3 \sin(\theta) + \beta_4 \cos(2\theta) + \beta_5 \sin(2\theta)\ldots \\ &= \beta_1 + \beta_2 \cos(-\theta) + \beta_3 \sin(-\theta) + \beta_4 \cos(-2\theta) + \beta_5 \sin(-2\theta) \\ &= \beta_1 + \beta_2 \cos(\theta) - \beta_3 \sin(\theta) + \beta_4 \cos(2\theta) - \beta_5 \sin(2\theta) \end{aligned} \tag{13}$$

$$\beta_3 \sin(\theta) + \beta_5 \sin(2\theta) = -\beta_3 \sin(\theta) - \beta_5 \sin(2\theta) \tag{14}$$

$$R_s(\theta) = \beta_{s1} + \beta_{s2} \cos(\theta) + \beta_{s3} \cos(2\theta) \tag{15}$$

Appendix C: The motion capture dataset

The *Motion Capture Dataset* (MCD) consists of 19 primitive action classes and 5 high-level scenarios. Overall, 10 subjects performed each action 20 times and each scenario 5 times.

Used terms

In the following, the main expressions/abbreviations are explained. These expressions will be used in the detailed action description.

- **SUBJECT**: The person performing the actions is called SUBJECT.
- **NPOSE**: This is the Neutral pose. The SUBJECT stands straight having both arms pointing downwards on the corresponding body-side. The foot are at the same height, next to each other, at shoulder width apart. The head is straight, looking forward.
- **START_POSITION**: The START_POSITION is defined as the position in which the user starts the action. In many cases this position can be defined on a fixed point. The SUBJECT is asked to stand on a marked area within the NPOSE, having the legs positioned in a comfortable way (shoulder width).

Actions

The subjects were asked to perform the actions in a natural smooth movement, following the provided descriptions.

- **STAND_TO_SIT**: The SUBJECT stands on the marked START_POSITION, having a chair behind him/her. The chair is positioned in such a way that the SUBJECT can sit down without moving the chair (If the chair is adjustable in height, the SUBJECT is asked to adjust it before). The STAND_TO_SIT action is then performed by the SUBJECT sitting down on the chair.
- **SIT_TO_STAND**: This action represents the reverse movements from STAND_TO_SIT, i.e. now the SUBJECT starts sitting on a chair and stands up ending in a neutral pose (NPOSE).
- **STEP_FORWARD**: The SUBJECT stands on the marked STARTING_POSITION and makes a natural step, using the right leg, to the front. After this step the left leg is positioned next to the right leg, i.e. the SUBJECT is ending in the NPOSE on step in front of the START_POSITION.
- **STEP_BACKWARD**: This action is similar to the STEP_FORWARD action. Instead of stepping forward, the SUBJECT steps backwards with his/her right leg. After the step the left leg is positioned at the same height as the right leg ending in the NPOSE - analog to STEP_FORWARD, but in this case one step behind the START_POSITION.
- **STEP_RIGHT**: The SUBJECT stands on the START_POSITION and steps with his / her right leg one step to the right. Following the left leg is positioned next to the right leg ending in a neutral pose (NPOSE) again.
- **STEP_LEFT**: This action is analog the STEP_RIGHT action, but the SUBJECT steps to the left. I.e. the SUBJECT steps, starting from the START_POSITION, with

the left leg one step to the left - followed by the right leg, which is positioned next to the left leg again - ending up in the NPOSE.

- **WALK_FORWARD**: The WALK_FORWARD action consists of a five step walk. Therefore the SUBJECT starts on the START_POSITION and steps five steps forward (right, left, right, left, right) - starting with the right leg. After the last step the SUBJECT stands on the right leg. Now the left leg is positioned at the same height as the right leg, ending up in the NPOSE. i.e. the beginning and end are similar to the STEP_FORWARD action.
- **WALK_BACKWARD**: This action represents the reverse movement of WALK_FORWARD, i.e. the SUBJECT walks five steps backwards. Therefore he/she starts at the START_POSITION and steps, starting with the right leg, five steps backwards (right, left, right, left, right). After the last step, the left leg is positioned at the same height than the right leg again (NPOSE).
- **TURN_RIGHT_90**: The turning actions represent full body turns. TURN_RIGHT_90 represents corresponding to the name a right turn of 90°. Hence the SUBJECT stands at the START_POSITION and turns 90° to the right and on the spot with the full body - ending in a neutral pose (NPOSE). This turn is performed at the same position, whereas small changes are accepted. The movements with legs are not specified, i.e. the SUBJECT can perform them freely (but in a natural way).
- **TURN_LEFT_90**: Analog to the TURN_RIGHT_90, this action represents a full body left turn. The SUBJECT starts at the START_POSITION and turns 90° to the left ending in the NPOSE (compare to TURN_RIGHT_90). The movements of the legs are free to the SUBJECT.
- **TURN_RIGHT_180**: Analog to TURN_RIGHT_90, but instead of turning 90° the SUBJECT has to turn 180° (Compare to TURN_RIGHT_90).
- **TURN_LEFT_180**: Analog to TURN_LEFT_90, but instead of turning 90° the SUBJECT has to turn 180° (Compare to TURN_LEFT_90).
- **APPROACH** : The SUBJECT has to approach an object - in this case a light switch. Hence the SUBJECT stands in the NPOSE in a comfortable position in front of the light switch and pushes it with the right hand. The action stops at the point where the SUBJECT touches the switch. The light switch is at a height of 1.2-meters. No movements of the legs is required, nevertheless the SUBJECT is not restricted to change the position of the legs.
- **PUSH**: Within this action the SUBJECT has to push an object. Two markers are attached on a tabletop (one marker near the SUBJECT, the other with more distance). The object is placed at the near marker and the SUBJECT stands at the START_POSITION in front of the table - such that he/she can reach both markers. The right hand touches the object already. To perform the action the SUBJECT pushes the object from the near to the more distant marker using the right hand. During the push, the object doesn't lose contact with the tabletop and its orientation is kept in a similar way (smaller changes are accepted).
- **PULL**: This action represents the reverse action to PUSH. I.e. the object is located at a distant marker and the SUBJECT has to pull it to the near marker using the right hand. (Please compare to PUSH in order to have a more detailed description).
- **GRASP**: The GRASP action represents an object grasp. Therefore the SUBJECT stands in front of a table within the START_POSITION (NPOSE). To perform the action the SUBJECT grasps the object. The action stops when the SUBJECT touches the object.
- **PLACE**: The PLACE action describes the placing of an object. Hence it can e.g. be executed after the GRASP position. The SUBJECT stands in front of a table with two markers (one marker left and one right). An object is placed on top of one marker and it has to be moved to the other marker. During the START_POSITION, the SUBJECT already has grasped the object. Now he/she has to move the object to the second marker. During this movement the object looses contact with the tabletop. The movement stops when the object is at the final position (SUBJECT still touches the object).
- **OPEN_DOOR**: The SUBJECT stands in the START_POSITION in front of a closed door (NPOSE). He/she has to open the door, in order to go through it. Which hand is used is free to the SUBJECT. Furthermore the leg movements are not described. The task for the SUBJECT is to open the door in a natural way.
- **CLOSE_DOOR**: Opposite to the OPEN_DOOR action, the SUBJECT has to close a door within this action. Hence, the SUBJECT stands in front of an open door and has to close it in a natural way (Compared to OPEN_DOOR).

Scenarios

- **DAF_OFF**: The SUBJECT started lying in the bed. He/she was asked to get up and start his/her morning routine as if it would be a free day.
- **GO_TO_WORK**: This scenario starts similar to the DAY_OFF scenario. However, this time the SUBJECT was asked to perform a morning routine as if he/she had to go to work.
- **PICK_A_SNACK**: The SUBJECT starts sitting on the couch. He/she is asked to get up and take a snack.
- **PICK_REMOTE**: The SUBJECT starts sitting on the couch again. Now, he/she is asked to get up and pick the remote controller for the television.
- **TIDY_UP**: Similar to the two scenarios above, initially the SUBJECT sits on the couch. He/she is asked to tidy up the apartment. Therefore all objects have been placed on defined initial positions. Corresponding to the initial positions, final positions have been defined. Hence, the subject had to move the objects from the initial position to the corresponding final position. However, the order was not given.

Appendix D: Derivation of the wave transient

In the following, we provide a mathematical derivation of the wave transient, for a complete derivation we direct the reader to the work presented in [57]. The initial equation is the dynamic field (1). Now, it is assumed that the field, which is used to generate the moving peak, has a local excitation (peak solution) (see a-solution [48]) and that the input signal $S(x, t) \equiv 0$. In order to generate movement, an asymmetric interaction kernel $w_a = w_e + w_0$, consisting of a symmetric kernel part w_e overlapped with an asymmetric function w_0, is developed. The shape of the function w_0, which is necessary to generate the movement, is determined in the following. By taking the previous assumptions as well as the asymmetric kernel w_a into account, the dynamic field equation (1) results into

$$\tau\dot{u}(x,t) = -u(x,t) + h + \int f(u(x',t))w_a(x-x')\mathrm{d}x'. \quad (16)$$

Assuming that there is an initial stable peak solution within the field at time $t = 0$, meaning $U(x) = u(x, 0)$. Thus, the excitation distribution, for any time instance $t > 0$, is given by

$$U(x,t) = U(x + \int_0^t v(\eta)\mathrm{d}\eta), \quad (17)$$

whereby $v(t)$ represents the velocity of the moving peak. Equation (17) can be used to calculate an equation providing information about the relation of w_0 and $v(t)$. Plugging (17) into the right side of (16) we obtain

$$\tau\dot{u}(x,t) = \tau U' \frac{d}{dt}(x + \int_0^t v(\eta)d\eta) = \tau U' v(t). \quad (18)$$

Plugging (17) into the left side of (16) results in

$$-U + \int_{-\infty}^{\infty} w_a(x,y)f(U(y))dy + h = \int_{-\infty}^{\infty} w_o(x,y)f(U(y))dy, \quad (19)$$

given the knowledge about the equilibrium solution under w_e is

$$\int_{-\infty}^{\infty} w_e(x,y)f(U(y))dy = U - h. \quad (20)$$

Finally, combining the left and right side we obtain

$$\tau U' v(t) = \int_{-\infty}^{\infty} w_o(x,y)f(U(y))dy. \quad (21)$$

It can be seen that the relation between w_0 and $v(t)$ is not as simple as may be expected. However, by setting $w_0 = p(t)w_e'$, and given the knowledge of (20) the complex relation simplifies to

$$v(t) = \frac{p(t)}{\tau}. \quad (22)$$

Here, $p(t)$ is a time-depending factor and w_e' the spatial derivation of the symmetric kernel part. Now, (22) allows to control the speed of the moving peak, whereas the shape of the kernel influences the direction. An example of this kernel is shown in Fig. 1(b) (black dashed line labelled with number 4).

References

[1] D. Feil-Seifer, M. J. Mataric, Defining socially assistive robotics, 9th International Conference on Rehabilitation Robotics (ICORR), 2005, 465–468

[2] A. Avci, S. Bosch, M. Marin-Perianu, R. M. Perianu, P. Havinga, Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey, 23rd International Conference on Architecture of Computing Systems (ARCS), 2010, 1–10

[3] L. W. Barsalou, W. K. Simmons, A. K. Barbey, C. D. Wilson, Grounding conceptual knowledge in modality-specific systems, Trends in cognitive sciences, 2003, 7(2), 84–91

[4] E. R. Smith, G. R. Semin, Socially situated cognition: Cognition in its social context, Advances in experimental social psychology, 2004, 36, 53–117

[5] G. Schöner, Dynamical systems approaches to cognition, Cambridge handbook of computational cognitive modeling, 2008, 101–126

[6] J. K. Aggarwal, M. S. Ryoo, Human activity analysis: A review, ACM Comput. Surv., 2011, 43(3), art. 16

[7] A. Bulling, U. Blanke, B. Schiele, A tutorial on human activity recognition using body-worn inertial sensors, ACM Computing Surveys (CSUR), 2014, 46(3), art. 33

[8] D. Lobato, Y. Sandamirskaya, M. Richter, G. Schöner, Parsing of action sequences: A neural dynamics approach, Paladyn, Journal of Behavioral Robotics, 2015, 6(1)

[9] E. Bicho, L. Louro, W. Erlhagen, Integrating verbal and nonverbal communication in a dynamic neural field architecture for human-robot interaction, Frontiers in Neuro robotics, 2010, 4(5), 1–13

[10] J. J. Gibson, The Ecological Approach to Visual Perception, Boston: Houghton Mifflin, 1979

[11] G. Semin, J. Cacioppo, Grounding social cognition: Synchronization, coordination, and co-regulation, in G. R. Semin, E. R. Smith (Eds.), Embodied grounding: Social, cognitive, affective, and neuroscientific approaches, Cambridge University Press, 2008

[12] G. R. Semin, E. R. Smith, Socially situated cognition in perspective, Social Cognition, 2013, 31(2), 125–146

[13] R. E. Shaw, E. Kadar, M. Sim, D. W. Repperger, The intentional spring: A strategy for modeling systems that learn to perform intentional acts, Journal of Motor Behavior, 1992, 24(1), 3–28

[14] M. A. Umilta, E. Kohler, V. Gallese, L. Fogassi, L. Fadiga, C. Keysers, G. Rizzolatti, I know what you are doing: a neurophysiological study, Neuron, 2001, 31(1), 155–165

[15] E. Kohler, C. Keysers, M. A. Umilta, L. Fogassi, V. Gallese, G. Rizzolatti, Hearing sounds, understanding actions: action representation in mirror neurons, Science, 2002, 297(5582), 846–848

[16] L. Fogassi, P. F. Ferrari, B. Gesierich, S. Rozzi, F. Chersi, G. Rizzolatti, Parietal lobe: from action organization to intention understanding, Science, 2005, 308(5722), 662–667

[17] N. Sebanz, H. Bekkering, G. Knoblich, Joint action: bodies and minds moving together, Trends in cognitive sciences, 2006, 10(2), 70–76

[18] J. J. Gibson, The Senses Considered as Perceptual Systems, Boston: Houghton Mifflin., 1966

[19] J. J. Gibson, E. S. Reed, R. Jones, Reasons for Realism: Selected Essays of J. J. Gibson, Resources for Ecological Psychology, L. Erlbaum, 1982

[20] A. Chemero, An outline of a theory of affordances, Ecological psychology, 2003, 15(2), 181–195

[21] K. S. Jones, What is an affordance? Ecological psychology, 2003, 15(2), 107–114

[22] T. A. Stoffregen, Affordances as properties of the animal environment system, Ecological Psychology, 2003, 15(2), 115–134

[23] C. F. Michaels, Affordances: Four points of debate, Ecological Psychology, 2003, 15(2), 135–148

[24] H. Heft, Affordances, dynamic experience, and the challenge of reification, Ecological Psychology, 2003, 15(2), 149–180

[25] W. H. Warren, Perceiving affordances: visual guidance of stair climbing, Journal of Experimental Psychology: Human Perception and Performance, 1984, 10(5), 683–703

[26] E.-J. Marey, Analyse cinématique de la marche, Comptes Rendus des Séances de lAcadémie des Sciences, Paris, XCVIII, 1884

[27] G. Johansson, Visual perception of biological motion and a model for its analysis, Perception & psychophysics, 1973, 14(2), 201–211

[28] W. H. Dittrich, Action categories and the perception of biological motion, Perception, 1993, 22(1), 15–22

[29] J. Lange, M. Lappe, A model of biological motion perception from configural form cues, The Journal of Neuroscience, 2006, 26(11), 2894–2906

[30] M. A. Giese, T. Poggio, Neural mechanisms for the recognition of biological movements, Nature Reviews Neuroscience, 2003, 4(3), 179–192

[31] M. A. Giese, Computational Principles for the Recognition of Biological Movements: Model-based versus feature-based approaches, Oxford University Press, 2005

[32] M. A. Giese, Biological and body motion perception, in J. Wagemans (Ed.), Oxford Handbook of Perceptual Organization, Oxford University Press, 2014

[33] M. A. Giese, Biological and body motion perception, Oxford Handbook of Perceptual Organization, 2014

[34] M. A. Giese, T. Poggio, Neural mechanisms for the recognition of biological movements, Nature Reviews Neuroscience, 2003, 4, 179–192

[35] R. Blake, M. Shiffrar, Perception of human motion, Annu. Rev. Psychol., 2007, 58, 47–73

[36] J. M. Zacks, S. Kumar, R. A. Abrams, R. Mehta, Using movement and intentions to understand human activity, Cognition, 2009, 112(2), 201–216

[37] M. Iacoboni, I. Molnar-Szakacs, V. Gallese, G. Buccino, J. C. Mazziotta, G. Rizzolatti, Grasping the intentions of others with one's own mirror neuron system, PLOS Biology, 2005, https://doi.org/10.1371/journal.pbio.0030079

[38] V. Gallese, A. Goldman, Mirror neurons and the simulation theory of mind-reading, Trends in cognitive sciences, 1998, 2(12), 493–501

[39] G. Rizzolatti, L. Fogassi, V. Gallese, Neurophysiological mechanisms underlying the understanding and imitation of action, Nature Reviews Neuroscience, 2001, 2(9), 661–670

[40] V. Gallese, C. Keysers, G. Rizzolatti, A unifying view of the basis of social cognition, Trends in cognitive sciences, 2004, 8(9), 396–403

[41] E. Oztop, M. Kawato, M. A. Arbib, Mirror neurons: functions, mechanisms and models, Neuroscience letters, 2013, 540, 43–55

[42] V. Gallese, L. Fadiga, L. Fogassi, G. Rizzolatti, Action recognition in the premotor cortex, Brain, 1996, 119(2), 593–609

[43] S. T. Grafton, L. Fadiga, M. A. Arbib, G. Rizzolatti, Premotor cortex activation during observation and naming of familiar tools, Neuroimage, 1997, 6(4), 231–236

[44] L. Fadiga, L. Fogassi, V. Gallese, G. Rizzolatti, Visuomotor neurons: Ambiguity of the discharge or motor perception?, International Journal of Psychophysiology, 2000, 35(2), 165–177

[45] M. Kellenbach, M. Brett, K. Patterson, Actions speak louder than functions: the importance of manipulability and action in tool representation, Journal of Cognitive Neuroscience, 2003, 15(1), 30–46

[46] C. B. Boronat, L. J. Buxbaum, H. B. Coslett, K. Tang, E. M. Saffran, D. Y. Kimberg, J. A. Detre, Distinctions between manipulation and function knowledge of objects: evidence from functional magnetic resonance imaging, Cognitive Brain Research,

2005, 23(2) 361–373

[47] M. A. Arbib, G. Rizzolatti, Neural expectations: A possible evolutionary path from manual skills to language, Communication & Cognition, 1996

[48] S. Amari, Dynamics of pattern formation in lateral-inhibition type neural fields, Biological Cybernetics, 1977, 27(2), 77–87

[49] S. A. Ellias, S. Grossberg, Pattern formation, contrast control, and oscillations in the short term memory of shunting on-center off-surround networks, Biological Cybernetics, 1975, 20(2), 69–98

[50] Y. Sandamirskaya, S. K. U. Zibner, S. Schneegans, G. Schöner, Using dynamic field theory to extend the embodiment stance toward higher cognition, New Ideas in Psychology, 2013, 31(3), 322–339

[51] A. Bastian, G. Schöner, A. Riehle, Preshaping and continuous evolution of motor cortical representations during movement preparation, European Journal of Neuroscience, 2003, 18(7), 2047–2058

[52] Y. Sandamirskaya, G. Schöner, An embodied account of serial order: How instabilities drive sequence generation, Neural Networks, 2010, 23(10), 1164–1179

[53] Y. Sandamirskaya, G. Schöner, Serial order in an acting system: A multidimensional dynamic neural fields implementation, IEEE 9th International Conference on Development and Learning (ICDL), 2010, 251–256

[54] Y. Sandamirskaya, M. Richter, G. Schöner, A neuraldynamic architecture for behavioral organization of an embodied agent, IEEE International Conference on Development and Learning (ICDL), 2011, 2, 1–7

[55] F. L. da Silva, Neural mechanisms underlying brain waves: from neural membranes to networks, Electroencephalography and clinical neurophysiology, 1991, 79(2), 81–83

[56] J. M. Horschig, J. M. Zumer, A. Bahramisharif, Hypothesis-driven methods to augment human cognition by optimizing cortical oscillations, Frontiers in Systems Neuroscience, 2014, 8(119)

[57] R. Menzner, A. Steinhage, W. Erlhagen, Generating interactive robot behavior: A mathematical approach, From Animals to Animats 6: Proceedings of the Sixth International Conference on Simulation of Adaptive Behavior, MIT Press/Bradford Books, 2000, 135–144

[58] I. Iossifidis, A. Steinhage, Controlling an 8 dofmanipulator by means of neural fields, International Conference on Field and Service Robotics, 2001, 1–7

[59] Y. Lu, Y. Sato, S. Amari, Traveling bumps and their collisions in a two-dimensional neural field, Neural Computation, 2011, 23(5), 1248–1260

[60] D. Roetenberg, H. Luinge, P. Slycke, Xsens mvn: Full 6dof human motion tracking using miniature inertial sensors, Technical report, XSENS TECHNOLOGIES, 2013

[61] A. M. Glenberg, What memory is for: Creating meaning in the service of action, Behavioral and Brain Sciences, Cambridge University Press, 1997, 20(1), 41–50

[62] L. W. Barsalou, Language comprehension: Archival memory or preparation for situated action?, Discourse Processes, 1999, 28(1), 61–80

[63] W. Prinz, A common coding approach to perception and action, Springer, 1990

[64] D. I. Perrett, M. H. Harries, R. Bevan, S. Thomas, P. J. Benson, A. J. Mistlin, A. J. Chitty, J. K. Hietanen, J. E. Ortega, Frameworks of analysis for the neural representation of animate objects and actions, Journal of Experimental Biology, 1989, 146(1), 87–113

[65] N. F. Troje, Reference frames for orientation anisotropies in face recognition and biological-motion perception, Perception, 2003, 32(2), 201–210

[66] V. Caggiano, L. Fogassi, G. Rizzolatti, J. K. Pomper, P. Thier, M. A. Giese, A. Casile, View-based encoding of actions in mirror neurons of area f5 in macaque premotor cortex, Current Biology, 2011, 21(2), 144–148

[67] E. Marinoiu, D. Papava, C. Sminchisescu, Pictorial human spaces: How well do humans perceive a 3d articulated pose?, IEEE International Conference on Computer Vision (ICCV), 2013, 1289–1296

[68] I. Bülthoff, H. Bülthoff, P. Sinha, Topdown influences on stereoscopic depth-perception, Nature neuroscience, 1998, 1(3), 254–257

[69] S. Schneegans, G. Schöner, A neural mechanism for coordinate transformation predicts pre-saccadic remapping, Biological cybernetics, 2012, 106(2), 89–109

[70] R. L. Williams II, Engineering biomechanics of human motion, Technical report, Ohio University, 2013

[71] J. A. Feldman, Four frames suflce: A provisional model of vision and space, Behavioral and Brain Sciences, 1985, 8(02), 265–289

[72] D. Marr, Vision: A computational investigation into the human representation and processing of visual information, WH San Francisco: Freeman and Company, 1982

[73] D. Marr, H. K. Nishihara, Representation and recognition of the spatial organization of three-dimensional shapes, Proceedings of the Royal Society of London, Series B, Biological Sciences, 1978, 200(1140), 269–294

[74] A. P. Georgopoulos, J. F. Kalaska, R. Caminiti, J. T. Massey, On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex, Journal of Neuroscience, 1982, 2(11), 1527–1537

[75] A. P. Georgopoulos, A. B. Schwartz, R. E. Kettner, Neuronal population coding of movement direction, Science, 1986, 233(4771), 1416–1419

[76] A. P. Georgopoulos, E. Karageorgiou, Understanding events: From perception to action, chapter Voluntary Arm Movements in the Motor Cortex, Oxford University Press, 2008, 229–254

[77] A. P. Georgopoulos, J. F. Kalaska, R. Caminiti, J. T. Massey, On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex, Journal of Neuroscience, 1982, 2(11), 1527–1537

[78] P. Dayan, L. F. Abbott, Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems, MIT Press, 2005

[79] D. I. Perrett, M. W. Oram, M. H. Harries, R. Bevan, J. K. Hietanen, P. J. Benson, S. Thomas, Viewer-centred and object-centred coding of heads in the macaque temporal cortex, Experimental Brain Research, 1991, 86(1), 159–173

[80] W. T. Newsome, C. D. Salzman, The neuronal basis of motion perception, Ciba Found Symposium, 1993, 174, 217–230

[81] M. Taira, S. Mine, A. P. Georgopoulos, A. Murata, H. Sakata, Parietal cortex neurons of the monkey related to the visual guidance of hand movement, Experimental brain research, 1990, 83(1), 29–36

[82] J. R. Flanagan, R. S. Johansson, Action plans used in action observation, Nature, 2003, 424(6950), 769–771

[83] C. L. Colby, J.-R. Duhamel, M. E. Goldberg, Ventral intraparietal area of the macaque: anatomic location and visual response properties, Journal of neurophysiology, 1993, 69, 902–902

[84] M. Oram, D. I. Perrett, Responses of anterior superior temporal polysensory (stpa) neurons to "biological motion" stimuli, Journal of Cognitive Neuroscience, 1994, 6(2), 99–116

[85] G. Mather, K. Radford, S. West, Low level visual processing of biological motion, Proceedings of the Royal Society of London, Series B: Biological Sciences, 1992, 249(1325), 149–155

[86] M. V. Peelen, P. E. Downing, The neural basis of visual body perception, Nature Reviews Neuroscience, 2007, 8(8), 636–648

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

[88] O. Lomp, K. Terzić, C. Faubel, J. M. H. du Buf, G. Schöner, Instance-based object recognition with simultaneous pose estimation using keypoint maps and neural dynamics, in Artificial Neural Networks and Machine Learning - ICANN 2014, Springer, 2014, 451–458

[89] Cosivina - Compose, simulate, and visualize neurodynamic architectures, An open source toolbox for Matlab (accessed: May 27th 2015), https://bitbucket.org/sschneegans/cosivina

[90] C. B. Holroyd, M. G. H. Coles, The neural basis of human error processing: reinforcement learning, dopamine, and the error-related negativity, Psychological Review, 2002, 109(4), 679–709

[91] M. Haruno, D. M. Wolpert, M. Kawato, Mosaic model for sensorimotor learning and control, Neural Computation, 2001, 13(10), 2201–2220

[92] D. M. Wolpert, K. Doya, M. Kawato, A unifying computational framework for motor control and social interaction, Philosophical Transactions of the Royal Society of London, 2003, 358, 593–602

[93] J. Demiris, G. M. Hayes, Imitation as a dual-route process featuring predictive and learning components: a biologically plausible computational model, Imitation in animals and artifacts, 2002, 327-361

[94] Y. Demiris, M. Johnson, Distributed, predictive perception of actions: a biologically inspired robotics architecture for imitation and learning, Connection Science, 2003, 15(4), 231–243

[95] Y. Demiris, B. Khadhouri, Hierarchical attentive multiple models for execution and recognition of actions, Robotics and Autonomous Systems, 2006, 54(5), 361–369

[96] Y. Demiris, G. Simmons, Perceiving the unusual: Temporal properties of hierarchical motor representations for action perception, Neural Networks, 2006, 19(3), 272–284

[97] E. Oztop, D. M. Wolpert, M. Kawato, Mental state inference using visual control parameters, Cognitive Brain Research, 2005, 22(2), 129–151

[98] J. Tani, Learning to generate articulated behavior through the bottom-up and the top-down interaction processes, Neural Networks, 2003, 16(1), 11–23

[99] J. Tani, M. Ito, Self-organization of behavioral primitives as multiple attractor dynamics: A robot experiment, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 2003, 33(4), 481–488

[100] J. Tani, M. Ito, Y. Sugita, Self-organization of distributedly represented multiple behavior schemata in a mirror system: reviews of robot experiments using RNNPB, Neural Networks, 2004, 17(8-9), 1273–1289

[101] J. Bonaiuto, E. Rosta, M. Arbib, Extending the mirror neuron system model, I, Biological Cybernetics, 2007, 96(1), 9–38

[102] E. Oztop, M. Kawato, M. Arbib, Mirror neurons and imitation: A computationally guided review, Neural Networks, 2006, 19(3), 254–271

[103] B. Akgun, D. Tunaoglu, E. Sahin, Action recognition through an action generation mechanism, in International Conference on Epigenetic Robotics (EPIROB), 2010

[104] Y. Yang, C. Fermüller, Y. Aloimonos, A cognitive system for human manipulation action understanding, in the Proceedings of the Second Annual Conference on Advances in Cognitive Systems (ACS), 2013, 109–124

[105] E. E. Aksoy, M. Tamosiunaite, R. Vuga, A. Ude, C. Geib, M. Steedman, F. Worgotter, Structural bootstrapping at the sensorimotor level for the fast acquisition of action knowledge for cognitive robots, in IEEE Third Joint International Conference on Development and Learning and Epigenetic Robotics (ICDL), 2013, 1–8

[106] F. Fleischer, V. Caggiano, P. Thier, M. A. Giese, Physiologically inspired model for the visual recognition of transitive hand actions, Journal of Neuroscience, 2013, 33(15), 6563–6580

[107] D. Newtson, Attribution and the unit of perception of ongoing behavior, Journal of Personality and Social Psychology, 1973, 28(1), 28–38

[108] J. M. Zacks, B. Tversky, Event structure in perception and conception, Psychological Bulletin, 2001, 127(1), 3–21

[109] M. M. Saylor, D. A. Baldwin, J. A. Baird, J. LaBounty, Infants' online segmentation of dynamic human action, Journal of Cognition and Development, 2007, 8(1), 113–128

[110] D. A. Baldwin, J. A. Baird, M. M. Saylor, M. A. Clark, Infants parse dynamic action, Child Development, 2001, 72(3), 708–717

[111] P. de Leva, Adjustments to Zatsiorsky-Seluyanov's segment inertia parameters, Journal of Biomechanics, 1996, 29(9), 1223–1230

[112] Y. Sandamirskaya, Dynamic neural fields as a step towards cognitive neuromorphic architectures, Frontiers in Neuroscience, 2014, DOI: 10.3389/fnins.2013.00276

[113] J. Zadny, H. B. Gerard, Attributed intentions and informational selectivity, Journal of Experimental Social Psychology, 1974, 10(1), 34–52

[114] D. Baldwin, J. Loucks, M. Sabbagh, Pragmatics of human action, in T. F. Shipley, J. M. Zacks (Eds.), Understanding events: From perception to action, Oxford series in Visual Cognition, Oxford University Press, 2008, 96–129

[115] B. Duran, Y. Sandamirskaya, Neural dynamics of hierarchically organized sequences: A robotic implementation, in 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids), 2012, 357–362