Dynamic Neural Fields as Building Blocks of a Cortex-Inspired Architecture for Robotic Scene Representation

Stephan K. U. Zibner, Christian Faubel, Ioannis Iossifidis, and Gregor Schöner

Abstract—**Based on the concepts of dynamic field theory (DFT), we present an architecture that autonomously generates scene representations by controlling gaze and attention, creating visual objects in the foreground, tracking objects, reading them into working memory, and taking into account their visibility. At the core of this architecture are three-dimensional dynamic neural fields (DNFs) that link feature to spatial information. These three-dimensional fields couple into lower dimensional fields, which provide the links to the sensory surface and to the motor systems. We discuss how DNFs can be used as building blocks for cognitive architectures, characterize the critical bifurcations in DNFs, as well as the possible coupling structures among DNFs. In a series of robotic experiments, we demonstrate how the DNF architecture provides the core functionalities of a scene representation.**

Index Terms—**Autonomous robotics, dynamic field theory (DFT), dynamical systems, embodied cognition, neural processing.**

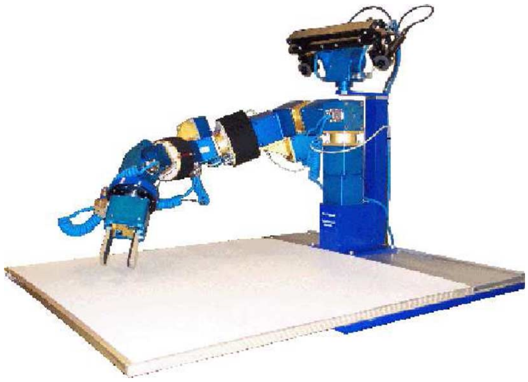

Fig. 1. Cooperative robotic assistant (CoRA) with an empty shared workspace in front of it, ready for adding objects to the scene.

I. Introduction

THE challenge and the pleasure of autonomous robotics research lies in its inherent interdisciplinarity. Autonomy requires that a robot be capable of acting based on its own sensory information. Any demonstration of an autonomous robot will therefore involve perceptual, planning, and motor control tasks, which must be interfaced and integrated. These tasks are interdependent. Not only does planning and motor control depend on perception, but also conversely robotic actions may modify the sensory stream and action plans may be aimed at obtaining particular perceptual information. The extraction of meaningful information about the robot's environment through perceptual systems is currently one of the major bottlenecks that holds back the development of autonomous robots.

For mobile robots, self-localization and mapping (SLAM) is a related problem, toward which much progress has been made over the last decades [1]. To generate goal-directed action that goes beyond moving to a particular location, robots need to have extended maps, in which objects are segmented [2], [3], and identified [4], [5]. To enable the reaching and grasping of objects, such a representation needs to include pose information about objects [6]. All three aspects of segmentation, identification, and pose estimation are currently underdeveloped. Even when laser scanners are used to capture the three-dimensional structure of the environment, extracting three-dimensional scene information is computationally very demanding [7] so that real-time updating of such three-dimensional scene information does not seem possible thus far. Another aspect of scene representation for robots is that objects [34] or object categories [9] must be learned on the fly from a small number of exposures. Our goal is to make progress toward the problem of scene representation for autonomous robotics by developing a neuronally inspired architecture that builds representations, enables their updating as the environment changes, and makes is possible to operate on scene representations through cued recall.

We focus on a particular component of the problem in an interaction scenario, in which a service robot shares a workspace with human users. Our cooperative robotic assistant CoRA [10] has a seven degree of freedom arm and an active stereo camera head, both mounted on a trunk that is fixed to a table (see Fig. 1). The table is the shared workspace between the robot and human users. A scene representation enables the robot to respond to user commands that refer to objects, object features, or locations on the table. For instance, in response to the command "hand me the red screwdriver," the system should be able to lo-

Manuscript received March 01, 2010; revised August 07, 2010, October 14, 2010; accepted December 15, 2010. Date of publication February 04, 2011; date of current version March 16, 2011. This work was supported by the German Federal Ministry of Education and Research within the National Network Computational Neuroscience—Bernstein Fokus under Grant FKZ 01GQ0951.

S. K. U. Zibner, C. Faubel, and G. Schöner are with the Institut für Neuroinformatik, Ruhr-Universität Bochum, 44780 Bochum, Germany (e-mail: stephan.zibner@ini.rub.de; christian.faubel@ini.rub.de; gregor.schoener@ini.rub.de).

I. Iossifidis is with the Institut Informatik, Hochschule Ruhr West Mühlheim, 44780 Bochum, Germany (e-mail: iossifidis@hs-ruhrwest.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAMD.2011.2109714

calize and segment the relevant object and estimate its pose sufficiently well to enable reaching. This is most effectively done based on a prior perceptual acquisition of the scene rather than by triggering a search at the time the command is received. This requires linking longer-term memories of objects and their features, obtained over multiple exposures to the objects, to the current layout of the scene. Using memory information that can also be updated is important because interaction with the robotic system happens under dynamic conditions in which objects may become occluded or get out of view because of the robot's cameras' limited field of view. In addition to a mechanism for longer-term memory, scene representation also requires a mechanism for working memory to handle such temporary occlusions while operating on an objects representation. Our neuronally inspired framework for scene representation will also support processes of selection such as when multiple red screwdrivers are in the scene, and tracking, such as when the screwdriver is handled by the human user.

In the spirit of the developmental approach to autonomous robotics, we derive ideas and constraints for the problem of scene representation from an analysis of how humans learn to achieve the associated tasks. When humans attend to a scene such as the workspace our robot CoRA, they process the scene sequentially. This sequentiality is due both to computational and physical constraints. Only a small number of objects can be in the perceptual foreground at any time. Moreover, objects are typically foveated for inspection. A saccade to foveate a new object is triggered on average every 300 ms. Visual information is not retained at a pictorial level between saccades [11], [12]. What visual information is retained across saccades depends on attention [13]–[15] as dramatically demonstrated by change blindness [16], in which major changes in an image go undetected if the changed locations are not attended to and the transient change signal, which would normally attract attention, is masked. Change blindness can be overcome by fixating on the changed item [17]. The visual representation of objects in a scene remains linked to space. Object discrimination is enhanced, for instance, when an object is presented in the same position in which it was first presented [18]. Conversely, providing scene context improves memory for object position [19]. The same position advantage disappears if the spatial configuration of other objects in the scene is scrambled, but not if the objects are coherently shifted [20]. This supports the notion that object information, both spatial and visual, is anchored in space.

To exploit these insights into how humans represent scenes, we build on a theoretical language, that has been used to model human spatial cognition. dynamic field theory (DFT)[1] [21], [22] originated as a theory of movement preparation [23], [24], but has recently been substantially extended towards higher-level cognition addressing visual working memory [25] and its development [26], [27], as well as feature binding [28]. The language builds on earlier work on how dynamical systems can be used to describe both human [29] and robotic behavior [30] in such tasks as target acquisition and obstacle avoidance. Dynamic neural fields (DNFs) enable the scaling of tasks to a more cognitive level such as working memory for the localization of targets [31] or the representation of obstacles [32]. Erlhagen *et al.* [33] used DFT to implement imitation learning and in Faubel and Schöner [34] a DFT architecture has addressed fast object learning and recognition.

A key assumption of DFT is that all behaviors are in stable states most of the time, making them immune to fluctuating sensory information and competing behaviors or representational states. Such stability arises not only in a control engineering sense through feedback loops, but also through internal loops of neuronal interaction. Behavioral flexibility then requires that states may be destabilized to bring about change of behavior. We will discuss the generic instabilities of DNFs and show how cognitive functions may emerge from these instabilities. This will enable us to use DNFs and their instabilities as buildings blocks for generating scene representations.

II. Architectural Principles

In this section, we briefly review core principles of DFT: the continuous metric spaces, over which neural fields are defined, their neural dynamics and stable states, as well as the relevant instabilities from which cognitive function emerges. We extend these principles toward multidimensional fields and DNF architectures by discussing the different possible forms of coupling among DNFs of varied dimensionality. Only those aspects of DFT are reviewed that matter for the architecture supporting scene representation. Therefore, we illustrate the concepts of DFT by referring to figures that are based on actual robotic experiments, which will be described in Section IV.

A. DNFs and Their Dimensionality

The set of possible perceptual and motor states of an embodied autonomous system may often be characterized by a number of continuously valued parameters. This is obvious for motor systems, in which movement parameters such as the Cartesian position of a tool point, joint velocity vectors, or the orientation of a robot head span relatively low-dimensional spaces of possible motor states. Perceptual states may be embedded in the two-dimensional visual array sampled by a vision system. Moreover, local feature detectors for color, orientation, or spatial wavelength may generate perceptual representations that may likewise be characterized by a limited number of feature dimensions. Below we will consider perceptual representations in which a feature dimension is combined with the two-dimensional coordinates of the visual array.

To represent objects in a scene as well as planned motor acts we employ the neural concept of activation. For every possible value along any of the relevant dimensions, an activation variable represents the presence of information by a large level of activation, the absence of information by a low level of activation. DNFs are the resulting distributions of neural activation defined as functions over such continuous, metric spaces. Fig. 2 illustrates a perceptual field, Fig. 3 a motor field. Localized peaks of activation are the units of representation in DNFs: When the activation level in a peak exceeds a threshold (conventionally chosen to be zero), the corresponding activation variables become effective input in whatever part of the system into which they couple. The location of such peaks along the continuous

[1] DFT is equally referred to as dynamic neural field theory (DNFT)

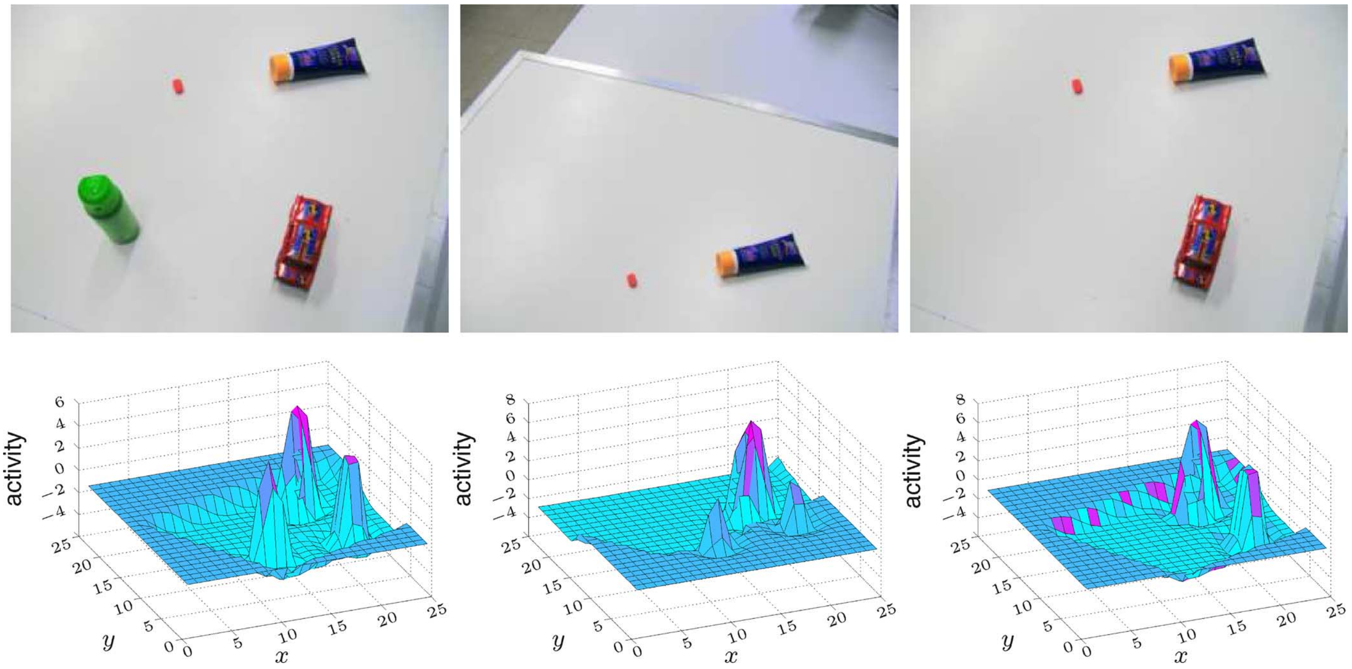

Fig. 2. Detection, working memory, and forgetting. These figures show a portion of the full architecture, the scene space field, demonstrating three basic instabilities. In the first step on the left, three perceived objects are visible in the current field of view. All three objects are represented in the field due to a detection decision, whereas a small perturbation on the table is not represented. Regions that are currently not in the visual range reside in a different regime and differ in the resting level. The figures in the middle show the field in a follow-up state, which is produced by changing the robot's gaze. Now, there is only a single object in the input image. Two working-memory peaks represent the other two objects. Due to different resting levels, both peaks are self-sustained. The figures on the right show the field state after the robot's gaze returned to its initial position. While two objects were outside the robot's gaze, one object was removed. After returning to the previous viewing angle, the input image only contains two objects. The working memory peaks in the previous field activity return to the region of lower resting levels. Since working memory cannot be sustained without additional visual input, the field forgets the missing object.

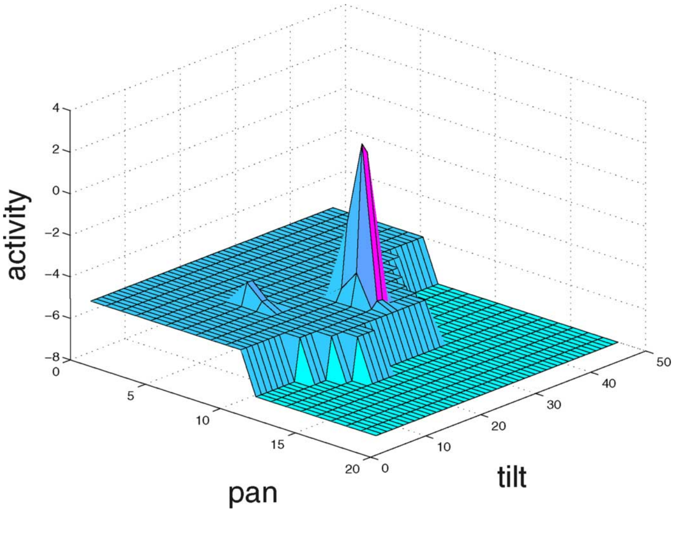

Fig. 3. Selection decision. The displayed motor selection field receives two competing inputs that are equally strong. Only one of the inputs is selected. A peak has built at its location, the other input is suppressed.

metric dimensions represents an estimate of the corresponding feature values and thus, encodes metric information about perceptual objects or motor plans.

How many dimensions are needed to characterize such objects or actions? Because of the computational cost of using DNFs with many dimensions, limiting the number of dimensions is an important concern. For instance, the control of a seven degree of freedom arm may at first seem to require a seven-dimensional space. There is no need, however, to such a high-dimensional DNF. Reaching may be characterized by the elevation and azimuth angles of the heading direction of the end-effector, which span a two-dimensional space. The tangential velocity of the end-effector may be encoded in a separate one-dimensional DNF [35]. An analytical solution to the inverse kinematics of the robot arm can then be used to expand an estimate of the desired motor state from these two low-dimensional spaces into the full seven-dimensional kinematic state of the arm. In other cases, the effector system itself is captured by a small number of dimensions. This is the case for motor control of the head used in our architecture, which comprises only the head pan and tilt angles. Similarly, the location of visual objects in the image plane may be captured by a two-dimensional field. A further reduction is not possible, however. If an object must be selected among a set of visible objects through an attentional process, then the two spatial dimensions must be coactivated. If selection were to occur separately in two one-dimensional field for each spatial dimension, then different objects may be selected along the horizontal compared to the vertical axis, leading to a mismatched spatial description (a so-called illusory conjunction). On the other hand, once an object has been selected, the motor commands for the pan and tilt angles of a camera head may perfectly well be represented separately each within a single one-dimensional field, as there is only one possible value along each dimension. As a more general rule, information can be kept separately in low-dimensional fields, as long as there is no need for associations of concurrent activation in multiple fields. From a practical view, lower dimensional fields are computationally much cheaper as their high-dimensional counterparts and can therefore have a better sampling of the continuous metric they represent.

To encode the combination of a single visual feature such as color with the two-dimensional visual array requires a three-dimensional DNF. As more feature dimensions are added, a combinatorial explosion threatens. This explosion may be avoided, however, if locations in multiple lower dimensional space-feature fields may be bound together along a shared dimension [28], [34], a trick we will discuss at the end of Section II-G.

B. The Dynamics of DNFs

The temporal evolution of patterns of activation in DNFs is generated from neuronal dynamics governed both by inputs and by neuronal interaction within a field. This interaction is structured such that the units of representation, localized peaks of activation, may emerge as stable activation states (or attractors). Local excitatory interaction stabilizes such peaks against decay, while global inhibition stabilizes peaks against diffusive broadening [36].

To formalize the neural dynamics, we describe the DNFs by activation variables $u(\vec{x},t)$ that are defined over the continuous metric dimensions, $\vec{x}$, and evolve in time t. The dynamic equation of such higher dimensional fields is analogous to the one-dimensional neural field dynamics first analyzed by Amari [36]

$$\begin{aligned}\tau\dot{u}(\vec{x},t) = &-u(\vec{x},t) + h + s(\vec{x},t)\\ &+ \int\cdots\int w(\vec{x}-\vec{x}')\theta(u(\vec{x}',t))d\vec{x}'. \quad (1)\end{aligned}$$

The first three terms set up the field as a temporal low-pass filter of input, $s(\vec{x},t)$. Based on these terms alone, the field relaxes toward the instantaneous stable state, $h + s(\vec{x},t)$ (as long as s is varying slowly enough).

Neuronal interaction (last term) is mediated by the nonlinear threshold function, $\theta(u(\vec{x},t))$, that typically has sigmoidal shape (0 below a threshold, u_θ, conventionally chosen to be zero, 1 above the threshold, with a more or less steep transition between these two limit cases). As a result, only sufficiently activated field locations contribute to neuronal interaction. The interaction kernel, $w(\vec{x}-\vec{x}')$, is positive (excitatory) for small distances between field locations, $\vec{x}$ and $\vec{x}'$, and negative (inhibitory) over larger distances. For more details, see (16) in Appendix A.

C. Dynamic Instabilities

DNFs as cognitive buildings blocks offer a set of operational regimes in which different stable states exist. They determine which tasks can be fulfilled by a field. These regimes may be characterized by studying the instabilities that occur when inputs or parameters of DNFs change. While the stability of peak solutions has been treated analytically for one- [36] and two-dimensional fields [37], higher dimensional fields have not been similarly well characterized analytically. We have been guided, nevertheless, by Amari's analysis in order to find the parameter settings at which peak solutions become stable in three-dimensional fields. The proof of their stability was then based on numerical simulation.

1) Detection Instability: The detection instability is the most elementary bifurcation and is at the origin of any supra-threshold peak. When input drives activation above threshold at any particular location, local excitatory interaction is engaged and destabilizes the subthreshold activation pattern. The peak "pulls itself up." The peak solution is qualitatively different from the subthreshold pattern of activation. This is obvious from the fact that the peak continues to be stable when input is again reduced: at intermediate levels of input, supra-threshold peaks and subthreshold patterns of activation coexist bistably. This stabilizes the peak in the face of fluctuating input. The qualitative change from subthreshold to peak solution may be thus used to represent a detection decision (see Fig. 2).

2) Selection Decisions and Fusion: In many situations a robot must select among multiple competing choices, for example, to orient its body or head toward one out of a number of salient objects. DNFs can organize such selection decisions. This fact has previously been exploited to account for neural and behavioral data on how humans select visual targets toward which they direct saccadic eye movements [38], [39] (see also Fig. 3). The inhibitory interaction is the key to selection. If the locations of multiple inputs are spaced adequately and inhibitory interaction is sufficiently strong, then an existing peak may inhibit peak formation at other stimulated locations. The sigmoidal nonlinearity creates an asymmetry of interaction: The selected site may inhibit competing sites, while the subthreshold activation at those sites does not contribute to interaction. Which location is selected thus depends on prior activation. Whichever site was able to generate supra-threshold activation first has the competitive advantage. The temporal order of stimulation is thus one important competitive factor. If multiple inputs arrive at the same time and have the same strength, then random fluctuations determine the outcome of the competition. The system is then multistable: a peak at any of the locations would be stable. This multistability persists when inputs differ in strength. As a result, the selection decision is stabilized: an initial selection is stable even if input fluctuations or deterministic changes of input begin to favor another location. This stability breaks down when the discrepancy in input strength becomes too large: at the selection instability, a peak at a location with weaker input becomes unstable and yields to a peak centered on the more strongly driven location.

3) Boost-Induced Detection and Selection: When localized input is too weak to induce a supra-threshold peak, a homogeneous boost to the field may drive it through the detection instability. A peak arises then at location that receives (weak) localized input. Selection may effectively be engaged by such a homogeneous boost as well, when multiple field locations are preactivated.

4) Self-Sustained Peaks as Working Memory and the Forgetting Instability: Amari derived conditions under which supra-threshold peaks of activation may persist in the absence of any localized input [36]. These conditions depend on the integral of the interaction kernel, $\int d^n x w(\vec{x})$, over varied domains (here, n, is the dimension of the field) as well as on the homogeneous resting level of the field, h. The kernel integral over the entire support of the kernel must be negative and the resting level must

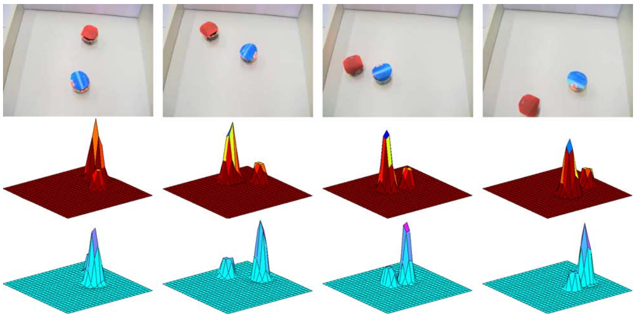

Fig. 4. Figure shows four snapshots of the tracking experiment (see Section IV-C) with the mobile robots. The red and the blue plots are slices of activation extracted from the *scene space–color field* for the corresponding robot. The red plot shows the activation for the red robot and the blue plot shows the activation for the blue robot.

be above the negative value of the (positive) maximum of the kernel integral. We have used this same logic for the two- and three-dimensional implementations of DNFs, and have evaluated the validity of these conditions in those higher dimensional cases.

Such a self-sustained peak can be thought of as a form of working memory [40], in which the position of the peak within the field encodes previously cued metric information. Working memory for metric information is useful to store objects that get out of the robot's current field of view, but whose spatial or feature parameters were previously estimated [31]. A working memory peak is volatile: it may be destabilized by competition. Such forgetting by interference may lead a previously stored metric value to be "forgotten." A controlled way of forgetting is to push the field through the forgetting instability, in which a lowering of the resting level destabilizes sustained peaks. This is illustrated in Fig. 2, in which a negative homogeneous boost effectively decreases the overall resting level h.

5) Multipeak Working Memory: It is possible to configure a DNF such that solutions with multiple localized peaks are stable. This happens most easily, if the interaction kernel has the shape of a Mexican hat, that is, if inhibition decreases again at sufficiently large distances between locations. The number of possible peaks within a field depends on the resting level h and the exact shape of the kernel (see Erlhagen and Bicho [21] for a discussion). In all cases, the number of peaks is limited by the fact that each peak has an overall inhibitory effect on the field. As such inhibition accumulates, the peaks ultimately become unstable. This fact has been exploited to account for the limited capacity of visual working memory in humans [41], [42].

6) Tracking: The stable supra-threshold peaks in DNFs are sensitive to changes in localized input. A moving localized input distribution is easily tracked by an associated peak. Even multi-item tracking is possible as has been shown in Spencer and Perrone [43]. Fig. 4 shows screenshots and field activity of such a multiitem tracking experiment.

D. Discrete Dynamic Neurons and Neural Assemblies

Under some circumstances, it is useful to think of peaks of activation as individual dynamic entities. The activation within a peak is then described by a single activation variable $u(t)$ and its dynamics

$$\dot{u}(t) = -u(t) + h + s(t) + w_{\text{exc}}\theta(u(t)). \qquad (2)$$

The local excitatory interaction that stabilizes peaks is now represented as self-excitation of this single activation variable, $w\theta(u(t))$. This dynamics has the analogous instabilities of detection and forgetting, so that a bistable regime with an "on" state (activation variable above threshold) and an "off" state (activation variable below threshold) is typical. We employ discrete activation variables to represent the presence of a stable peak irrespective of the exact location of the peak ("peak detector"). This is achieved by projecting the supra-threshold activation integrated across a whole field onto a single, dynamic node, which is thus driven through a detection instability if there is at least one peak in the field.

Multiple, competing activation variables of this nature may be used to represent activation patterns, in which the individual entities, represented by the different variables, are not in an obvious way embedded in an underlying continuous space (e.g., discrete object labels). We employ such ensembles of discrete activation variables

$$\dot{u}_l(t) = -u_l(t) + h + s(t) + w_{\text{exc}}\theta(u_l(t)) - w_{\text{inh}}\sum_{l' \neq l}\theta(u_{l'}(t)) \qquad (3)$$

to represent different objects as a whole (through "labels" of the objects). The connectivity of such ensembles of activation variables with self-excitation and global inhibition leads to a "winner takes all" behavior in which only one activation variable may have positive activation, while all others are inhibited below threshold.

E. Memory Traces as a Form of Long-Term Memory

A possible mechanism for long-term memory consists of modulating the level of activation in a field based on memory traces of prior patterns of activation. Such memory traces are represented in a separate field defined over the same dimension, but with its own dynamics that evolves on a much slower timescale τ_{pre}. A dynamics of low-pass filtering any supra-threshold activation in the original neural field generates such a memory trace

$$\tau_{\text{pre}}\dot{p}(\vec{x},t) = \alpha_{\text{peak}}[-p(\vec{x},t) + \theta(u(\vec{x},t))] \cdot [\lambda_{\text{b}}\theta(u(\vec{x},t)) + \lambda_{\text{d}}(1 - \theta(u(\vec{x},t)))]. \qquad (4)$$

Here, $p(\vec{x},t)$, is the memory trace activation. A peak detector, α_{peak}, is implemented with a discrete activation variable that receives as input the summed and thresholded activation, $\theta(u(\vec{x},t))$, of the field. Memory traces are only updated when peaks build up in the DNF. The memory trace builds up with the rate λ_{b} at active field sites $(\theta(u(\vec{x},t)) = 1)$ and decays with rate λ_{d} at inactive sites $(1 - \theta(u(\vec{x},t)) = 1)$. The global timescale τ is the same for the field that creates the memory trace. The timing of building and forgetting memory traces

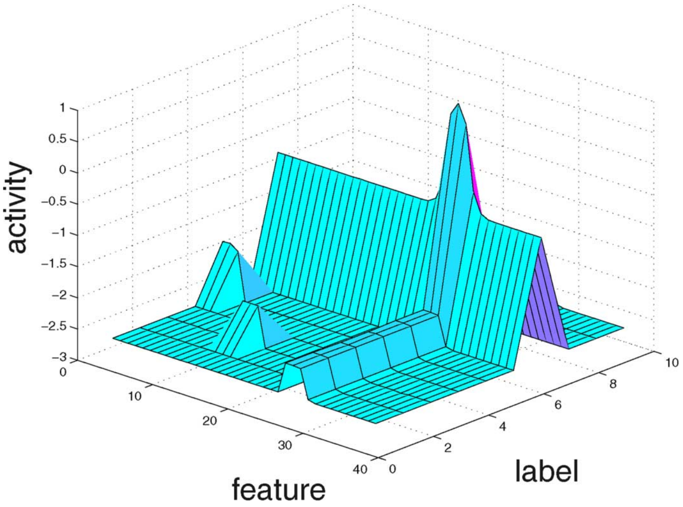

Fig. 5. Ridge inputs and memory traces. This figures shows the object–color field in a state of receiving two lower dimensional inputs of label and color information. The overlap of both ridges creates a peak at the intersection. The preshape dynamic of this field deposits memory traces for every arising peak in this field. Two deposited traces originate from previous peaks, a third one is generated by the currently active peak.

is controlled through the terms λ_{b} and λ_{d}. See Fig. 5 for an example of deposited preshape in a DNF.

If such a memory trace is conversely coupled as additive input into the DNF, of which it receives input, then the memory trace mechanism has the properties of Hebbian learning: previously active field locations are preactived by the memory trace they have laid down and are thus easier to again activate in the future. In this form, the memory trace preshapes the DNF, biasing it toward previously experienced patterns of activation. Below, we show how peaks may be generated from a preshaped field in a way that effectively reinstates the previously experienced state ("recall"). The concept of a memory trace and its role to preshape representations has been used to account for the role of behavioral history in movement preparation [23] and infant perseverative reaching [24], [44], as well as a host of other forms of long-term memory [45].

F. Coupling of Fields

DNF architectures consist of sets of coupled DNFs, some of which are directly linked to sensory surfaces or receive some form of preprocessed input, while others are directly linked to motor systems and specify particular stable states of motor control. Because the dynamics of the individual DNFs have attractor states, these persist when fields are coupled. Only when a field goes through an instability, it is sensitive to inputs. Thus, the component fields in DNF architectures can be analyzed individually with respect to their stable states and instabilities, treating coupling as a form of input (this would fail only if instabilities were to occur simultaneously in two coupled fields, which is not a generic case).

Coupled fields may differ in the nature and number of field dimensions as well as in the metrics of neuronal interaction. In the following paragraphs different possible mappings are elaborated in some detail. We begin with coupling among fields of the same dimensionality, then consider projections from higher to lower dimensional fields and finally the mapping from lower to higher dimensional fields.

1) Coupling of Fields With the Same Dimensionality: The coupling of two DNFs with the same dimensionality is straightforward: the output of one DNF, $\theta(v(\vec{x},t))$, provides localized input to the target neural field $u(\vec{x},t)$

$$\begin{aligned}\tau\dot{u}(\vec{x},t) = -u(\vec{x},t) + h + s(\vec{x},t) + w_{uv}\theta(v(\vec{x},t))\\ + \int\cdots\int w_{uu}(\vec{x}-\vec{x}')\theta(u(\vec{x}',t))d\vec{x}'. \quad (5)\end{aligned}$$

The mapping may be spatially modulated through a Gaussian convolution kernel, $w_{uv}(\vec{x})$, that is homogeneous along the fields' dimensions

$$\begin{aligned}\tau\dot{u}(\vec{x},t) = -u(\vec{x},t) + h + s(\vec{x},t)\\ + \int\cdots\int w_{uu}(\vec{x}-\vec{x}')\theta(u(\vec{x}',t))d\vec{x}'\\ + \int\cdots\int w_{uv}(\vec{x}-\vec{x}')\theta(v(\vec{x}',t))d\vec{x}'. \quad (6)\end{aligned}$$

In numerical implementation, the discrete resolution of coupled fields may need to be adjusted by down- or up-sampling (e.g., by linear interpolation). To avoid sampling errors, down-sampled fields must be smoothed.

2) Coupling Higher Dimensional to Lower Dimensional Fields—Integration: A higher dimensional field, v, may be homogeneously coupled to a lower dimensional field, u, if there is at least one shared dimension between the two fields. Activation in the higher dimensional field is integrated along the nonmatching dimension and used as weighted input along the matching dimension. For instance, the equation for mapping a three-dimensional field onto a one-dimensional field with x as the matching and y, z as nonmatching dimensions reads

$$\begin{aligned}\tau\dot{u}(x,t) = -u(x,t) + h + s(x,t)\\ + \int w_{uu}(x-x')\theta(u(x',t))dx'\\ + w_{uv}\int_y\int_z \theta(v(x,y,z,t))dydz. \quad (7)\end{aligned}$$

Kernels defining the more complex weight mapping from one field to another may also be defined.

3) Coupling Lower Dimensional to Higher Dimensional Fields—Ridges, Slices and Tubes: To homogeneously couple a lower dimensional field to a higher dimensional field, the two also need to share at least one dimension, which is expanded along the nonmatching dimension(s). For instance, mapping a one-dimensional field to a two-dimensional field

$$\begin{aligned}\tau\dot{u}(x,y,t) = -u(x,y,t) + h + s(x,y,t)\\ + \iint w_{uu}(x-x', y-y')\\ \times\theta(u(x',y',t))dx'dy'\\ + w_{uv}\theta(v(x,t)) \quad (8)\end{aligned}$$

creates input ridges into the two-dimensional field (see Fig. 5). When a one-dimensional field is mapped onto a three-dimensional field

$$\begin{aligned}\tau\dot{u}(x,y,z,t) = &-u(x,y,z,t)+h+s(x,y,z,t)\\ &+\iiint w_{uu}\theta(u(x',y',z',t))dx'dy'dz'\\ &+w_{uv}\theta(v(x,t))\end{aligned} \tag{9}$$

with $w_{uu} \cong w_{uu}(x-x', y-y', z-z')$, the resulting inputs have the form of slices of the three-dimensional field within which input is constant. A two-dimensional field mapped onto a three-dimensional field

$$\begin{aligned}\tau\dot{u}(x,y,z,t) = &-u(x,y,z,t)+h+s(x,y,z,t)\\ &+\iiint w_{uu}\theta(u(x',y',z',t))dx'dy'dz'\\ &+w_{uv}\theta(v(x,y,t))\end{aligned} \tag{10}$$

with $w_{uu} \cong w_{uu}(x-x', y-y', z-z')$, creates input tubes into the three-dimensional field, along which input is constant (see Fig. 6 for slices and tubes).

4) Coupling of DNFs With Single Discrete Neurons: A special case is the coupling of higher dimensional fields onto single activation variables. In this case, the field must be integrated over all dimensions to produce a scalar value that can then be weighted and used as input to the dynamics of the single activation variable. An example is the dynamics of the peak detector

$$\tau\dot{u}_{\text{peak}} = -u(t)+h+\int v(\vec{x},t)d\vec{x}+\theta(u(t)). \tag{11}$$

In the reverse direction, a single discrete activation variable may be coupled to a DNF by providing homogeneous input to the field. This may be employed, for instance, to induce boost-induced detection instabilities as explained in Section II-C3.

5) Coupling DNFs With No Matching Metrics: To couple two DNFs that do not share any common metrical dimension, supra-threshold field activation may be passed from one field to another with an arbitrary mapping. Such a mapping may be either hand-wired or be learned using learning rules such as Hebbian learning. For a more detailed explanation of arbitrary mappings and a DNF architecture in which such maps play a key role see Sandamirskaya and Schöner [46].

G. Functions Achievable by Coupling

By coupling DNFs we can effectively close the loop between the sensory surfaces and motor control and enable a robot to generate robust behavior. But coupling can do more than projecting a stabilized percept onto motor decisions. Coupling makes it possible to link DNFs that represent higher level percepts such as object labels to related lower level representations of a feature dimension. Coupling may create hierarchies of DNFs (see Section III). The higher a field is positioned in such a hierarchy, the further it is removed from the sensory or motor surface. Such higher fields are invariant under more transformations of sensory input. This makes such fields well suited to erecting working or long-term memory. Higher fields are thus the suitable substrate for cognition.

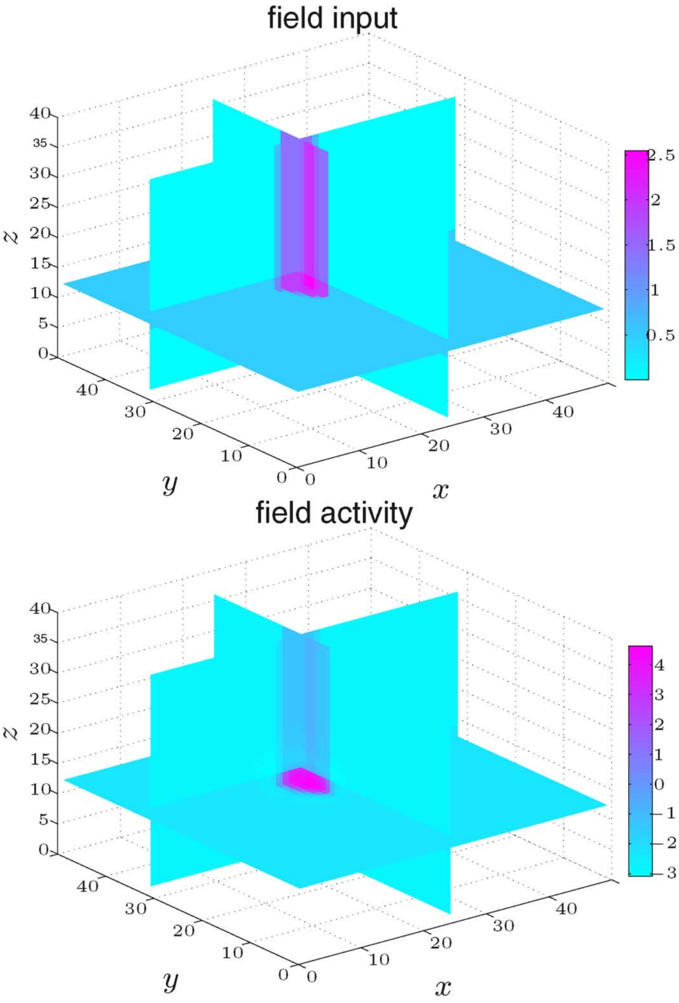

Fig. 6. Slices and tubes. The top plot shows spatial tube and color slice input to the three-dimensional *scene space–color field* for the green deodorant. In the field plotted below the overlap of the tube and slice input lead to a self-sustained working memory peak, that represents the green color of the deodorant and its spatial location on the table.

1) Higher Dimensional Representations: Multiple lower dimensional feature representations can be combined into higher dimensional representations by linking separate DNFs along one or multiple shared dimensions. Visual or motor space is a natural choice for such shared dimensions, because it reflects the physical reality that different feature dimensions are bound through the spatially localized objects, from which they emanate. This mechanism has been used in a DNF model of binding in visual working memory [47]. A dynamic field representing spatial working memory with high precision provides ridge input into space-feature fields (space–color, space-orientation) that are broadly tuned to space. Such coupling along visual space selects appropriate feature conjunctions. Similarly, the same basic mechanism was exploited in an object recognition system, in which multiple label-feature fields were coupled along the label dimension [34].

2) Association: In the previous example, the association of space with a feature dimension reflected immediately the stimulus. In principle, however, DNFs of suitable dimensionality may represented arbitrary associations [48]. This happens in our architecture in a three-dimensional field during the build-up of working memory for scene items, because at the scene level feature information is not available for all spatial locations but only for currently attended ones. The extracted feature is represented in a one-dimensional field that sends slice input into a three-dimensional field representing the feature dimension combined with the two-dimensional spatial layout of the workspace. The spatial selection provides tube input at the currently selected spatial position. At the location where the tube input overlaps with the slice input a peak builds up in the three-dimensional space-feature field. This peak is a working memory peak that in this way retains the cued association across occlusions and multiple saccades (see also Fig. 6).

Similarly, when an arbitrary label is learned, the association mechanism is again at work. A one-dimensional feature field sends ridge input into a two-dimensional label-feature field. The user activates a label to be associated, which provides ridge input along the label dimension. At the location where both ridges overlap a peak builds which leads to the creation of a memory trace, providing the system with a long-term memory for the label-feature association (see Fig. 5).

3) Preshape Mechanisms: Input that does not by itself induce a detection instability and an associated peak *preshapes* the field. Such preshape raises the propensity for peaks to arise at sites that are preactivated by preshape. Models of movement preparation have proposed that memory traces of supra-threshold patterns of activation may in effect bring about preshape in a given field [23], [24], [44]. Preshape need not necessarily, however, arise from a long-term memory mechanism. Preshape may also be used to implement top-down mechanisms of biased competition by providing a competitive advantage to a spatial location or feature value based on a decision at a higher level of scene representation. In our architecture, this happens when a selection peak builds up in the scene representation field providing a competitive advantage in the attention selection module. As the scene representation is a three-dimensional space-feature field and the attention selection is only defined over the two-dimensional space there is also a generalization in this mechanism. This generalization is a feature of coupling from higher to lower dimensional fields.

The preshape mechanism can also be used to generate categorical responses. In combination with the boost-induced detection mechanism a field with preshape and relatively weak input will build a peak at the position of the preshape and not at the position of the sensory input.

4) Triggering Transitions Between Operational Regimes and Sequencing: Peak detectors may be used to raise or lower the homogeneous resting level of fields and thus to switch them into different operational regimes. We make use of this mechanism to switch off the selection peak within the attention selection field. When a peak is detected within the label-feature field, this signals that an association has either been learned or recognized. The peak detector then triggers a negative boost to the selection field so that the selection peak is destabilized. This signal from the peak detector relates to what was termed the condition-of-satisfaction neuron in the DNF-based implementation of serial order [49].

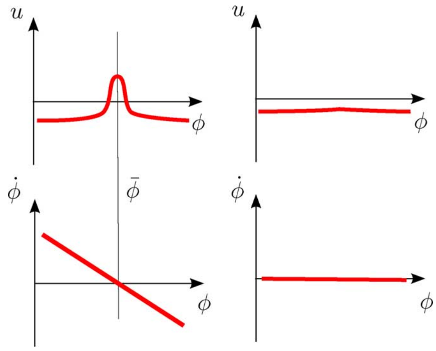

Fig. 7. Mapping the field activity to attractors of a dynamical system for a motor variable. The upper row shows two one-dimensional fields defined over a motor metric like movement direction. The field on the left has a single supra-threshold peak, the field on the right has no such peak. Below each field is plotted the resulting dynamical system for the motor variable that implements the motor metric. In the left column, the presence of a peak induces a fixed point attractor, marked by the zero-crossing of the rate of change with a negative slope. The stability of the attractor, determined by the negative slope, decreases with decreasing amount of supra-threshold activation. In the right column, the absence of a peak leads to a flat dynamics without attractors. All values of the motor variable are then marginally stable fixed points.

In our architecture sequencing emerges from the signal provided by the peak detector that switches off the selection peak and a negative preshape into the attention selection field coming from active working memory in the scene representation.

5) Coupling to Motor Behavior: Within the dynamical system approach, movement is controlled by defining dynamical systems with attractor states at the desired configuration, ψ, of the motor system

$$\tau\dot{\phi} = -\alpha(\phi - \psi). \tag{12}$$

How may the activation of a DNF be mapped onto such an attractor dynamics?

If one thinks of the field activation as a probability distribution, the desired configuration is determined by the theoretical mean of the distribution

$$\bar{\phi} = \frac{\int \phi \times \theta(u(\phi))d\phi}{\int \theta(u(\phi))d\phi}. \tag{13}$$

A direct approach would be to set $\psi = \bar{\phi}$ in (12). This leads to a division by zero, however, if the field does not have supra-threshold activation. In contrast to a real probability distribution, a DNF is not necessarily normalizable. Scaling the strength, α, of the attractor, with the total activation, $\alpha = \int \theta(u(\phi))d\phi$, gives rise to an elegant solution of this problem

$$\tau\dot{\phi} = \left(-\int \theta(u(\phi,t))d\phi\right)\left(\phi - \frac{\int \phi \times \theta(u(\phi))d\phi}{\int \theta(u(\phi))d\phi}\right) \tag{14}$$

can then be simplified to

$$\tau\dot{\phi} = \left(-\int \theta(u(\phi,t))d\phi\right)\phi + \int \phi \times \theta(u(\phi,t))d\phi \tag{15}$$

without any division (see Fig. 7 for an illustration of this mapping).

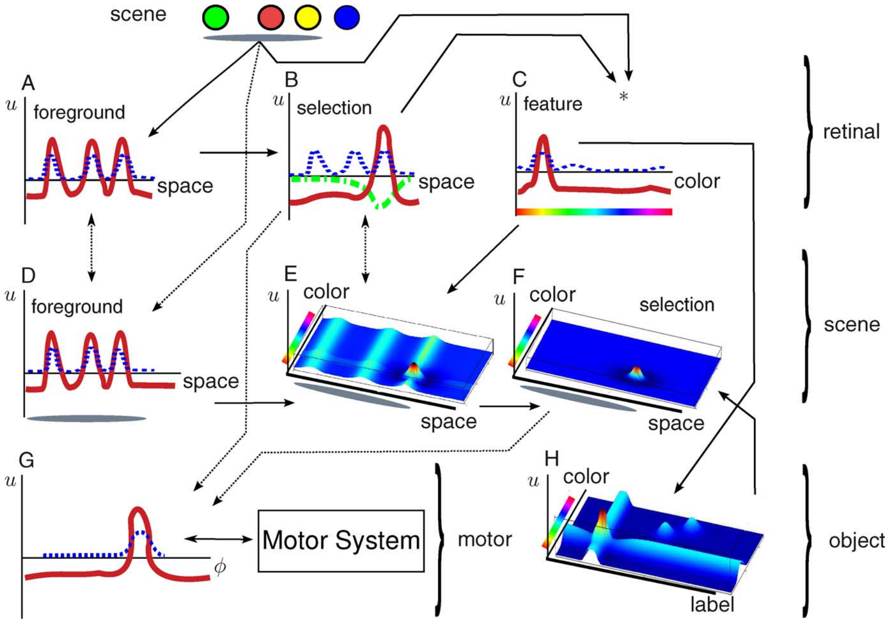

Fig. 8. Shows an overview of the cortex-inspired architecture. Space is illustrated as one-dimensional. All components are arranged in affiliation to the four levels. The current gaze is represented in the *retinal* and *scene level*, as well as in the scene by a gray oval. Information transfer between components is illustrated by two kinds of arrows: the solid ones represent information transfer without transformations, whereas their dotted counterpart implies an included reference frame transformation. In all field sketches, the solid red line represents the field activity. Blue dotted lines are standard excitatory input and green dashed and dotted lines are inhibitory input. The system is shown in a state, in which one of four possible objects is already scanned and stored in the associative *scene space–color field* E. The three leftmost objects are contained in the current *retinal space field* A as well as in the *scene space field* D. Both are mutually coupled and receive additional input from visual preprocessing. The scanned object is inhibited in the *retinal space selection field* B, which receives input from A. The still-active selection leads to a color extraction, which is represented in the *retinal color field* C. B is also coupled to a *motor selection field* G, which in turn is linked to the motor system. Finally, B, C, and D are all coupled to E. The recall of memorized object locations and features is done in an associative *scene space–color selection field* F similar to E, but which is, unlike E, configured to allow only a single peak at a time. The *label-color field* H contains space-independent long-term memory associations between object labels and color hue. Features are provided by *retinal color field* C, whereas labels are defined by the user. Learning an object triggers the next execution of an object scan. The right-most object in the scene was never seen by the system. There is no evidence of this object throughout the system.

III. A Cortex-Inspired Architecture for Scene Representation

In order to make the concepts developed in Section II do some real work, we propose a cortex-inspired architecture for scene representation that will be implemented for our autonomous robot CoRA. From the robotic scenario we may derive some constraints that simplify this task. Only the shared workspace, the table in front of CoRA, needs to be represented because the robot will operate only on objects positioned on that table. Relevant objects are restricted in size because only objects that fit into the robot's gripper shall be represented. The transformation from retinal coordinates into a table-based allocentric representation is known at all times and given by the head configuration of the robot. With these constraints we can evaluate our architectural approach in a real scenario, which can nevertheless be simplified enough so that technical issues do not distract from our focus on the interaction between the different DNFs. In this same spirit, the single feature dimension, color, used for object representation as well as the associated processes of feature extraction and selection are place holders for more complex DNF-based object recognition systems such as the label-feature field approach [34] or the combined pose estimation and recognition system [50]. Similarly, the transformation between different frames of reference is done algorithmically here, although such transformation can, in principle, be performed with the same class of neural dynamic mechanism [51].

The architecture for scene representation consists of ten DNFs that are coupled in a structured way. We order these fields into four levels based on the functionality and the degree of invariance of each field (see Fig. 8). These levels may be loosely associated with areas of the human cortex. The *retinal level* is the level closest to the sensory surface with the highest degree of variance. This level could be viewed as a functional description of visual cortex. The next level we refer to as the *scene level*, where spatial information is represented in a table-based allocentric reference frame and object feature information is linked to spatial location. The *scene level* may be associated with the lateral intraparietal area (LIP) [52]. The infero–temporal cortex is associated with visual object representation [53], which in our architecture corresponds to the *object level*. Finally, at the *motor level* head motion is represented, thus closing the action perception loop. This level may be neuronally associated with the frontal eye fields [54], the superior colliculus [55], and the brain stem [56].

Although organized into levels, the fields are mutually coupled and it is from this interaction that cognitive function emerges. The architecture will not function properly if the connections between levels are blocked. If, for example, the scene representation level were decoupled from the retinal level it would not receive any feed-forward input and would not do much work at all. Similarly the retinal level would not function properly, if both stabilizing input from the scene representation as well as the inhibitory input that supports selection were missing. We preconfigure each field in order to put it into the desired dynamic regime such as, for example, the regime in which multiple working memory peaks are possible or in which a single location is selected. The stability of these states makes it possible to establish and test the dynamics of each field individually. This makes it possible to tune and debug the architecture in a systematic way.

A. Retinal Level

The basic processes of segmentation, attentional selection, and low-level feature extraction take place at the *retinal level*. At this level, sensory input is highly variant: Every head movement modifies the incoming sensory stream in addition to any changes that may occur in the scene itself. Three DNFs are at work at the *retinal level*. One multipeak field, the *retinal space field*, receives input from a simple saliency computation and feeds its output into a second field that is set up for selection, the *retinal space selection field*. A third feature field, the *retinal color field*, receives as input a hue histogram computed from a hue color map of the input image.

1) Visual Preprocessing: The first stage of visual input computes a simple saliency image by calculating on- and off-center responses on the intensity and on two opponent color channels. Because the size of objects in our scenario is limited to fit into the robot's gripper, we can tune the size of the difference of Gaussians filters to approximately fit the size of the objects. All the responses are summed up with equal weights into a single saliency image. This simplified version of Itti *et al.* saliency computation [57] is sufficient in our scenario, but could of course be extended to include more features like orientation maps and multiple scales as in their original proposal. In addition to this saliency map we compute a hue color map that serves as input to the *retinal color field*.

2) Retinal Space Field: The result of the saliency computation is mapped onto a two-dimensional DNF that allows for multiple peaks. These peaks represent the locations of objects. The kernels of this field approximately match the Difference-of-Gaussians filters used for computing the saliency image. The output of this field is a normalized and stabilized version of the saliency image. Objects smaller than what the robot is able to grasp will not produce peaks and thus go undetected.

The output of this DNF is fed into the *retinal space selection field*. Furthermore, it is spatially transformed into the allocentric table representation and used as input to the *scene space field* on the *scene level*.

The representation of input in retinal coordinates is affected by head movements. The retinal position of a static object constantly changes as long as the head moves. During movement the two-dimensional field has to be capable of tracking multiple objects. This task becomes easier if a stabilized representation of object positions in the allocentric reference frame is used to generate predictions where objects are in retinal coordinates when the head moves in a certain direction. This information is projected back from the *scene space field* on the *scene level*.

Objects outside the table region should not produce peaks in the retinal space representation. We use knowledge about the table geometry to transform what is represented in the *scene space field* into retinal coordinates and thus project the *scene space field* into the retinal representation as a source of preshape. As a result, the resting level within in the retinal representation is substantially different in those regions of the image into which the table surface falls. Outside of this region, no peaks will build due to the low resting level there.

3) Retinal Space Selection Field: In order to extract a feature representation of the spatial locations, these must be brought into foreground. A *retinal space selection field* receives input from the *retinal space field*. The selection peak represents a single selected spatial location, which is then used for computing a local feature histogram. The output of the *retinal space selection field* is also projected to the *scene space–color field*. Furthermore it projects to the *motor selection field* so that selected items are centered on the camera.

4) Retinal Color Field: Extracting features of a possible object at the correct retinal position is achieved with the help of the *retinal space selection field*. Once a selection is active, the selection field's output can be used to mask all irrelevant retinal regions. Regions that pass the mask are used to extract a color hue histogram for a specific object. This histogram is used as input into a feature field [see Fig. 8(c)], that uses the detection instability to represent dominant object colors within the selected spatial region. The field output is fed as ridge input into the label-feature field at the *object level* and fed as slice input into the space-feature field at the *scene level*.

B. Scene Level

At the *scene level*, spatial position is represented with respect to an allocentric reference frame attached to the table. As the robot is attached to the table, this allocentric frame is at the same time an egocentric reference frame for the robot. Three different fields are at work, the *scene space field* that represents only the spatial configuration of the scene, and two fields representing the object feature color over a two-dimensional spatial map. One of these three-dimensional space-feature fields, the *scene space–color field*, acts as working memory field for attended spatial locations and their associated feature, the other, the *scene space–color selection field*, is used for selecting one of those working memory peaks based on other additional input that preshapes this selection field. This preshape comes from other cognitive modules. All three fields are defined in a reference frame related to the table. This representation is thus invariant to head movements, but because it receives input from the retinal level is still able to track moving objects. These fields may be loosely associated with the "where" path of processing in area MT and with links to more object related properties in hippocampal areas.

1) Scene Space Field: The *scene space field* is a multipeak spatial representation of object locations with invariance relative to the robot's own motion. The field receives spatially transformed input from the preprocessed visual information and from the *retinal space field*. This field has two important functional roles. On the one hand, by being invariant to the robot's own head movement, it provides the system with a mechanism for spatial working memory in which self-sustained peaks represent the locations of objects that are out of sight. On the other hand, self-stabilized peaks in this field keep track of objects that are in the current field of view. The field operates at two different resting levels: The area outside the current view is at a resting level that allows for multiple self-sustained peaks while the area within the field of view is at a lower resting level that enables only self-stabilized, but not self-sustained peaks, which therefore vanish when they lose support from input (see Fig. 2). Objects that disappear because of the robot's head movement are thus represented by working memory peaks, whereas objects that disappear from the sensory surface, because, for instance, someone takes them out of the field of view, are no longer represented.

The output of this spatially invariant representation is used to track the peaks in the *retinal space field* by predicting their future position given a planned head movement. That output is also used as tube input into the *scene space–color field*. The latter allows the *scene space–color field* to continuously track spatial changes while maintaining working memory for the space–color associations.

This is implemented through continuous coupling to the *scene space field* that provides the tube input that represents the spatial locations of the objects. Note that objects outside of the viewing angle cannot be updated, if an object is moved while it is outside this angle, it appears like a new object, once the viewing angle returns to this object. The old association is removed because the tube input, that sustains this association, moved.

2) Scene Space–Color Field: To create an associative working memory peak of object position and color in this field, three fields contribute their outputs. On top of the spatial tube input from the *scene space field*, the *retinal space selection field* provides a single item tube input giving only one spatial location within the *scene space–color field* an extra boost. Only at that location with the extra boost the slice input along the color dimension from the *retinal color field* leads to the build-up of a working memory peak. Once the selection in the *retinal space selection field* is released, the slice input of the *retinal color field* disappears as well and the working memory peak is solely supported from the tube input form the *scene space field*.

The output of the scene representation field is generalized to a purely spatial representation and transformed back to the retinal reference frame and used as inhibitory input to the *retinal space selection field*. The output is also fed directly into the *scene space–color selection field*.

3) Scene Space–Color Selection Field: Similar to the *retinal space selection field* we need a mechanism to bring items to the foreground that are internally represented. The *scene space–color field* provides no functionality to isolate a single stored association. A selection decision must therefore be made in a second three-dimensional field, the *scene space–color field*, which receives as input the space–color association, but operates in a selection regime. This field receives a copy of the sigmoidal working memory peaks in the scene representation as input. Additional inputs may be either lower dimensional spatial or feature cues, which may be gathered through user interaction. These inputs have the familiar form of tubes and slices and preshape the selection field. The overlap of scene memory and broad cue input increases the probability of selection for all those stored objects that correspond to the spatial or feature cue. The selection decision of the scene recall field selects the most appropriate object. If two or more objects are identical or share the correspondence to a cue, one of these is randomly selected.

Bringing an item into foreground ultimately means to make a spatial decision, because space is the only representation on which motor commands and thus meaningful action can be exerted. The three-dimensional data is generalized to a spatial output. This output is then used as input to a two-dimensional *motor selection field*.

C. *Object Level*

At the object level, the feature representation varies neither with spatial transformation in the scene nor with head movement. The representation at this level is the one with highest invariance. Feature input from the scene is only provided when an object has been actively selected in the *retinal space selection field*. As a result, this representation is only updated when an object has been actively selected.

1) Object Label-Color Field: The *object label-color field* is a two-dimensional association (see Fig. 5) field representing the hue color along one dimension and discrete labels along the other dimension. The feature input comes from the *retinal color field*, which only produces peaks when an item has been selected by the *retinal space selection field*. When users provide a label information, the ridge input along the labels overlaps with the ridge input from the feature dimension and a peak builds up in the *label-color field*. This peak leads to the build up of a long-term memory trace of this association. Once this memory trace is created it preshapes the association field so that provided matching feature input builds a peak in the label-color field without having to specify the label. This recognition mechanism is basically a single feature version of the label-feature field approach [34] and can be easily extended to multiple features.

D. *Motor Level*

At the *motor level*, head movements are planned in angular coordinates and motor signals are generated. Both the *retinal space selection field* as well as the *scene space–color selection field* can specify head movement. In principle they both may provide active peaks at the same time. This happens, for example, if a user provides input for a cued recall during the scanning of the scene. A selection decision mechanism is thus also needed at the motor level. Because of the high accuracy of the low-level servo controllers of the real hardware there is no explicit proprioceptive representation of the joint angles. Such a representation could be easily added, it would require the definition of another field representing the read out from the encoder values of the joints.

1) Motor Fields: The *motor selection field* (see also Fig. 3) receives input that is mapped from the *retinal space selection field* and from the *scene space–color selection field*. These mappings are only approximations and are explained in the Appendix D. The output from the two-dimensional *motor selection field* is projected onto two separate one-dimensional fields with sharper kernels that represent the pan and tilt values of the camera. These are the *motor pan field* and the *motor tilt field*. Peaks in those two fields set attractors in the corresponding attractor dynamics of the pan and tilt angles as explained in Section II-G5. The rates of change are send to the hardware interface of the head joints.

E. Sequence Generation

Once an item has been brought into foreground and its features have been extracted and stored both as working memory in the *scene space–color field* at the *scene level* and as long-term memory in the *label-color field*, the selection may be released and a new item should be selected. During learning this transition to a new item is triggered by the user providing label information for the current object. When an object is recognized, the transition happens autonomously.

The sequentiality in processing comes from two sources. First a peak detector mechanism at the level of the *label-color field* sends a negative boost to the selection field, so that it goes from the selection instability mode into the no-peak solution. A peak in the *label-color field* represents that an association has been learned and is taken as the condition-of-satisfaction for bringing an object into feature space and into the long-term representation. Second, negative preshape is sent back from the space–color representation at the *scene level*, effectively reducing the propensity for a peak to build at a spatial location which had already been examined earlier on and for which an active working memory is maintained at the *scene level*.

IV. Results

To evaluate our architecture, we conduct several experiments that probe different cognitive functionalities: the build-up of the scene representation; tracking spatial changes in the scene; updating the representation when objects are removed; keeping working memory representations of objects when they get out of view; and updating their representations when they become visible again; object recognition; recall of spatial information in response to a cue about the label of a long-term memory item and head movement toward the cued item.

A. Build-Up of the Scene Representation

The first experiment is a demonstration of the autonomous build-up for multiitem scenes. Three objects are placed on the workspace in front of the robot: a red toy car, a blue pack of tissues, and a green can of deodorant spray. See Fig. 9 for a basic setup. The task is to build up the scene representation by selecting each of the objects, one at a time. Once an item has been selected, its color represented in the *retinal color field* must be associated with the spatial location represented in the *scene space field*. This association will be stored in the *scene space–color field* and a long-term memory will be created as soon as the user provides a label signal. Once this label signal has been given, the next item is selected. When the robot brings an object into foreground, it orients its head toward this object so that the object is centered in the field of view of the camera. Peaks in the *retinal space field* and in the *retinal space selection field* must track the changes induced by the head movement. In total this experiment tests the following basic instabilities and field couplings: multiitem working memory both in the two-dimensional *scene space field* and in the three-dimensional *scene space–color field*, detection and selection decisions at the *retinal level*, the association mechanism at the *scene* and *object level*, setting attractors from peaks at the *motor level* and tracking of spatial changes in the *retinal level* when the head moves.

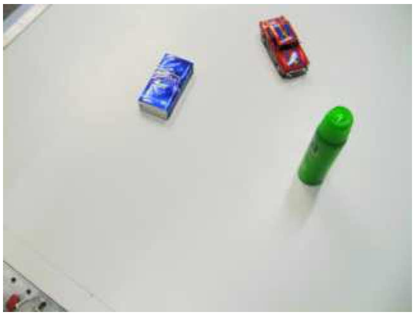

Fig. 9. Object setup. This is a saved camera image from the robot's left camera showing a basic setup of three objects: a blue pack of tissues on the left, a green can of deodorant spray placed on the right, and a red toy car in the middle. The whole setup is roughly placed around the center of the workspace.

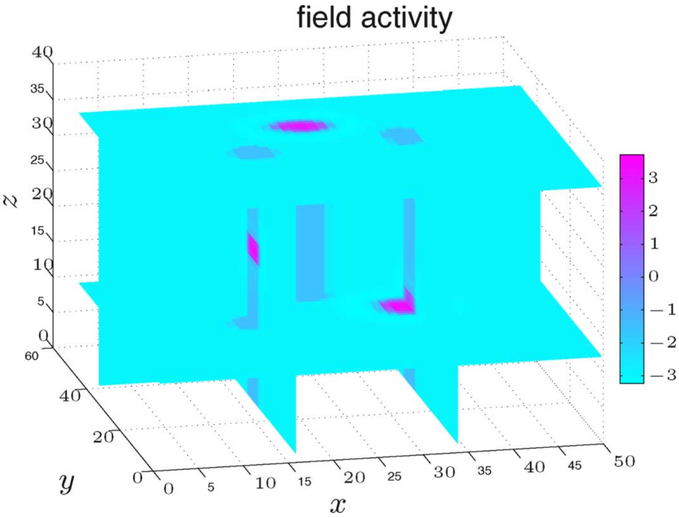

Fig. 10. Slice plot of the scene representation field from experiment 1. This figure shows a set of crossed slices from the three-dimensional volume of field activity in the *scene space–color field*. Dimensions x and y represent spatial information whereas dimension z spans the feature space for color hue. The displayed activity is the result of a scanning sequence. The *scene space–color field* contains three regions of supra-threshold activity referring to the three scanned objects. The regions correspond to the position in space and the extracted color hue of each object. The red toy car resides in the upper range of the color hue due to its cyclic nature, whereas the blue pack of tissues and green deodorant spray occupy regions further down the color hue. The tube inputs that sustain all three associations can be seen throughout the volume.

1) Results: The system was able to generate the scene representation as well as the long-term memory for object labels. See Fig. 10 for the resulting state of the scene representation, containing three working memory peaks, one for each object space–color association. The experiment was successfully repeated for different numbers of objects, different objects, and configurations. Small object distances were no issue due to separation along the feature dimension in the three-dimensional field. Head movements were compensated by the described predictions sent from the stabilized allocentric representation of object positions.

B. Head Movement and Working Memory and Updating

The second experiment is a test of working memory for space–color associations and also tests the updating when objects are removed from the scene. Objects are removed in two different configurations of the robot, once when the robot is attending the scene object and once when the item is out of view and maintained as working memory. Additionally we tested short occlusions by covering an object for a short moment. This experiment tests the multi-item working memory mechanism on the *scene level*, the forgetting instability and the continuous update of the scene. The setup for this experiment consists of three objects: a red toy car, a green can of deodorant, and a blue tube of sunscreen. See Fig. 2 for the setup. The task is once again to store scene information in the scene representation. The robot's head is then moved upwards until two out of three objects disappear from the current camera image. The deodorant spray is then removed from the workspace and the head returns to its original orientation. Then the blue tube of suncream is removed while it is visible.

1) Results: The two objects that get out of sight because of the head movement were both represented as working memory peaks (see the plot in the middle of Fig. 2). When the head returned to the initial view, now without the deodorant, the peak representation for this object vanished (see the plot on the right in Fig. 2). Removing the second object from the scene also led to a removal of the peak in the *scene space–color field*. In contrast covering the third remaining object shortly with a sheet of paper did not affect the representations in the *scene space–color field*.

C. Multiobject Tracking

In order to test the tracking capability of scene representation in a systematic and reproducible way, we use two small robotic platforms (E-Puck[2]). Each of them is marked with a different color. They are put into the scene and the scene build-up is started. Once the scene representation for this static scene has been created, they are switched on in the default Braitenberg obstacle avoidance behavior of Vehicle 2a [58] that comes already implemented on the E-Puck platforms. The robots start to move around randomly and because of the relative short line of sight of their infrared sensors (around 40 mm) they come relatively close (10 mm) during the course of the experiment. Note that while tracking the scene there is no active selection of a single object and therefore no head movement occurs while tracking the scene.

1) Results: The system was capable of tracking spatial positions of the robots and it maintained the correct color associations for the robots. See Fig. 4 for recordings from the tracking experiment.

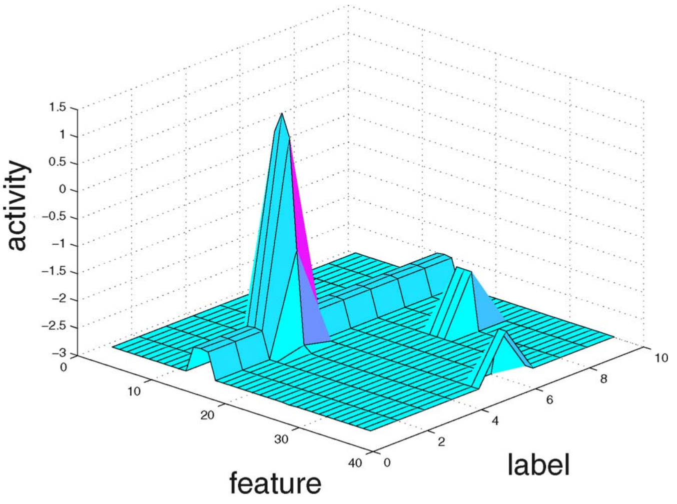

Fig. 11. Recognition of an object by overlapping preshape and a ridge color input.

D. Object Recognition

The fourth experiment demonstrates the recognition mode of the *label-color field*, the categorical response of a field preshaped by memory traces and how this recognition smoothly integrates into the build-up of scene representation. Again three objects are placed on the table and a first object is learned. Once its long-term memory label-color association is learned it is removed from the scene so that its working memory representations in the *scene space–color field* vanishes. Then the object is placed into the scene again.

1) Results: As there was no more spatial inhibition coming from the *scene space–color field* for this formerly learned object it was again selected at some random moment during the ongoing build-up of the scene representation. When the object, for which long-term memory had been deposited, was selected, a peak in the label-color field arose due to the overlap of the feature input with the preshape from the long-term memory trace (see Fig. 11). Once this peak was established the standard switching mechanism came into play and the next object was attended. The user did not have to specify the label again.

E. Cued Recall

In the cued recall experiment, the cue is given by providing label information, thus providing input along the label in the *label-color field*. This creates a peak in the *label-color* field at the learned color and the color-dimension then preshapes the *scene space–color selection field*. This field selects the working memory item from the *scene space–color field* that has most overlap with this color representation. Once a peak has formed in the *scene space–color selection field* as it projects to the *motor selection field* a head movement to center the cued location is activated.

1) Results: We tested cued recall with different objects and varied locations on the table. The robot successfully attended all objects when they were cued. They were not always perfectly recentered due to the approximation of the mapping from table

[2] http://www.e-puck.org/

to head joint coordinates (for more detail on this approximation please refer to the Appendix D). Essentially the robot always brought the items back into the camera view, the centering could then be driven by visual servoing as it happens during the build-up of the scene.

V. Discussion

A. Summary

We have presented an approach to scene representation that is inspired by what is known about how humans visually scan scenes, retaining spatial and feature information across fixations. The neuronally based theoretical language of DFT is at the core of this approach. We have shown how DNFs of varying dimensionality and functionality can be coupled to achieve the target cognitive functions of detection, selection, working memory, and tracking. With an architecture of ten coupled DNFs we have demonstrated how a robot may autonomously build a scene representation grounded in real-world vision data obtained from its cameras. The center piece of the architecture is a three-dimensional DNF that provides the system with working memory for associations between feature values and two-dimensional visual space. These associations can be established sequentially, one by one. Spatial change of the visual configuration is tracked and updated in real-time. In a set of experiments we have demonstrated the core functions of scene representation including the autonomous creation and continuous updating of elements of the scene representations as well as cued recall from an object long-term memory in response to a user command.

The concepts and models have been used in a separate line of research to account for human behavioral and neural data on looking, visual and spatial working memory, discrimination, and change detection [25], [28], [39], [45]. The robotic scene representation architecture that we built here within this framework may help address a number of technical issues by inheriting the autonomy, stability, and integrated nature of human visual cognition. The system is pervasively autonomous in the sense that a continuous process evaluates the neural dynamics, which react to sensory input as needed, including sensory input generated by the robot itself. Discrete events at which objects are detected, decisions are made, and memories are created emerge autonomously from that dynamics. Without a need for specific interrupt mechanisms, our system is naturally linkable to online changes of sensory information. Such autonomy requires that all functional states in the system are stable states. That fact, in turn, enables the system to work with relatively raw and low-level sensory information, which is noisy, fluctuates and drifts in time. Our system thus lowers the demands made by the scene representation on the perceptual channels. Finally, the uniform theoretical language of attractor neural dynamics provides the theoretical foundation and practical method for system integration. There is no additional level of algorithmic system integration. Once the DNF architecture has been designed, all integration has been achieved. In contrast, typical solutions using more conventional approaches draw on different methodologies such as visual preprocessing, probabilistic methods, or finite state machines. Their integration requires a specific effort at the level of the overall programming of the agent. Given the exemplary nature of our robotic demonstrations, only limited empirical evidence for the integrative power of DFT was provided in this project, however.

Our implementation of the architecture on a robotic platform achieved approximate real-time behavior. This is a direct demonstration that the use of even three-dimensional fields in a midsize model is not yet in any practical sense constrained by computational power. In the long run, optimization of the computational substrate for convolutions, the main time-limiting operations within the architecture, may further expand the range that can be reached within this approach (see, e.g., Dudek and Hicks [59]).

B. Relation to Saliency-Based Models of Attention and the Psychophysics of Scene Representation

Salience-based models of visual attention share a number of features with our approach. The sequential sampling of the visual array in our system, for instance, is somewhat similar to the inhibition of return mechanism in saliency-based models of attention [57]. In these models, activation is generated over a spatial map that reflects stimulus salience. The level of activation controls where the focus of attention is positioned. Locations that have been selected for attention are inhibited lowering the likelihood that they will be selected again. Most implementations of saliency maps for guiding attention do not address how eye movements affect such a map by shifting the camera plane relative to the environment (for an exception see Fix *et al.* [60]). The selection for attention happens, instead, in a fixed reference frame. Even in robotic implementations that have the potential for eye or head movements, saliency maps are typically based on keeping gaze fixed [61]. In a robotic attention model that addresses the effect of head and eye movement [62], previously attended regions are represented in an egocentric frame. In this map, attended regions are transiently activated and that activation decays over time. The system presented here goes beyond these approaches in several ways. First, we build a stable neuronal representation of space, which keeps track of shifts of the reference frame when head movements occur. This makes it possible to achieve the stability required to sustain periods of working memory and to stabilize selection decisions over variable delays, while at the same time enabling updating through a coupling to the sensory stream. The representation also encodes feature information, so that it is richer than a pure salience representation. It is therefore capable of providing an interface through the feature dimension with long-term memories of object features. This capability is highlighted by the cognitive task of cued recall.

A limitation of our current implementation is that we do not include visual transients as significant signals that control looking. This may be mended in the future by adding transient detectors into the input stream. Curiously, this makes that our approach reflects the phenomenon known as “change blindness” in the literature on human scene understanding [63]. When visual transients are masked, human observers routinely fail to detect major changes in a visual scene (such as an entire sizable object appearing or disappearing from the scene).

Change blindness illustrates how human scene understanding is based on cognitive rather than purely perceptual processes: the scene is largely constructed in the mind, not derived online from current sensory input. As in human observers, our system will update the representation of a changed item only when that item is currently activated and in the foreground, not when the item is merely in working memory.

C. Outlook

We have presented only the core principles and a very simple implementation of the proposed principles and architecture that employed limited and simplistic feature dimensions. The framework of DFT provides, however, a natural interface between scene representation and other systems implemented with the same framework. This will be exploited in the future to augment the functionality of our architecture. We have, for instance, been able to achieve competitive performance in single-shot object learning with a system based on DNFs for only three feature dimensions [34]. Using intrinsically more complex features, we achieved object recognition that was invariant under translation and rotation of objects [50]. This led to competitive performance both within the COIL benchmark as well as in scenes that required nontrivial segmentation due to partial occlusion. Linking to a model of change detection in visual working memory [25] may enable the system to autonomously decide when to update elements of its representation. Similarly, linking to a DNF model of how spatial language is grounded in vision [64], [65] may provide the system a user interface capable of interpreting and generating spatial cues such as in the phrase “hand the object to the left of the red screwdriver.”

The present architecture already contains elements of behavioral organization [66], that is, of the rule-based sequential activation of appropriate states. Such behavioral organization is required as objects are scanned one after another, the boost in the label-feature field leads to free recall, and the emerging feature representation is associated with a label. Dynamical systems principles are available to make that form of behavioral organization completely autonomous [66].

Appendix A
Kernels

Because the field is homogeneous, the interaction kernel is the same at all sites of the field and the kernel is symmetric. Mathematically the effect of the interaction kernel can be computed by convolving the nonlinear output function with the interaction kernel. Typically the kernel $w(\vec{x})$ is of the form of a Gaussian excitatory profile with constant global inhibition ($w_{\text{exc}} \neq 0; w_{\text{g}} \neq 0; w_{\text{inh}} = 0$), or with broader local inhibition, also modeled with a Gaussian profile ($w_{\text{exc}} \neq 0; w_{\text{g}} = 0; w_{\text{inh}} \neq 0$), or with a combination of both global and local inhibition ($w_{\text{exc}} \neq 0; w_{\text{g}} \neq 0; w_{\text{inh}} \neq 0$)

$$w(\vec{x}) = w_{\text{exc}} \frac{1}{\sigma_{\text{exc}}\sqrt{2\pi}} \exp\left(-\frac{(\vec{x}-\vec{x}')^2}{2\sigma_{\text{exc}}^2}\right) + w_{\text{inh}} \frac{1}{\sigma_{\text{inh}}\sqrt{2\pi}} \exp\left(-\frac{(\vec{x}-\vec{x}')^2}{2\sigma_{\text{inh}}^2}\right) + w_{\text{g}}. \quad (16)$$

Appendix B
Visual Processing Parameters

Before applying center-surround filters, the input image was downsampled from size 320 × 240 to 80 × 60

$$\text{On-center kernel}: \ \sigma = (2,2)$$
$$\text{Off-center kernel}: \ \sigma = (6,6).$$

Appendix C
Field Parameter Values

Retinal Space Field (size 80 × 60)

$$h = -1.0,\ \tau = 3,\ w_{\text{exc}} = 3$$
$$\sigma_{\text{exc}} = (2,2)$$
$$w_{\text{inh}} = -6,\ \sigma_{\text{exc}} = (4,4),\ w_{\text{g}} = -0.009.$$

Retinal space selection field (size 80 × 60)

$$h = -2.0,\ \tau = 3,\ w_{\text{exc}} = 5$$
$$\sigma_{\text{exc}} = (2,2)$$
$$w_{\text{g}} = -0.3.$$

Retinal color field (size 36)

$$h = -7.0,\ \tau = 5,\ w_{\text{exc}} = 3$$
$$\sigma_{\text{exc}} = 1$$
$$w_{\text{g}} = -0.5.$$

Scene space field (size 100 × 100)

$$h = 2.0,\ \tau = 4,\ w_{\text{exc}} = 8.4$$
$$\sigma_{\text{exc}} = (2,2)$$
$$w_{\text{inh}} = -8.9,\ \sigma_{\text{exc}} = (5,5),\ w_{\text{g}} = -0.01.$$

Scene space–color field (size 50 × 50 × 36)

$$h = -2.2,\ \tau = 3,\ w_{\text{exc}} = 5.1$$
$$\sigma_{\text{exc}} = (1,1,0.5)$$
$$w_{\text{inh}} = -4,\ \sigma_{\text{exc}} = (3,3,3),\ w_{\text{g}} = -0.002.$$

Scene space–color selection field (size 50 × 50 × 36)

$$h = -2.2,\ \tau = 10,\ w_{\text{exc}} = 1$$
$$\sigma_{\text{exc}} = (1,1,1)$$
$$w_{\text{g}} = -0.01.$$

Object label-color field (size 10 × 36)

$$h = -5,\ \tau = 10,\ w_{\text{exc}} = 2$$
$$\sigma_{\text{exc}} = (\text{discrete}, 1)$$
$$w_{\text{g}} = -0.2.$$

Motor selection field (size 80 × 180)

$$\begin{aligned} h &= -3,\ \tau = 10,\ w_{\mathrm{exc}} = 5 \\ \sigma_{\mathrm{exc}} &= (2,2) \\ w_{\mathrm{g}} &= -0.04. \end{aligned}$$

Motor tilt field (size 80)

$$\begin{aligned} h &= -7.0,\ \tau = 10,\ w_{\mathrm{exc}} = 1.3 \\ \sigma_{\mathrm{exc}} &= 1 \\ w_{\mathrm{g}} &= -0.4. \end{aligned}$$

Motor pan field (size 180)

$$\begin{aligned} h &= -7.0,\ \tau = 10,\ w_{\mathrm{exc}} = 1.3\ \sigma_{\mathrm{exc}} = 1 \\ w_{\mathrm{g}} &= -0.4. \end{aligned}$$

Appendix D
Scene–Motor Transformation

Consider $\vec{p} = (x, y)$ as two-dimensional position of an object and $\vec{c} = (c_x, c_y, c_z)$ the shift of a camera system with two joints pan and tilt. Let $x_s = x + c_x$ and $y_s = y + c_y$ be be the object position in a camera-centric coordinate system. Transforming x_s and y_s to polar coordinates yields

$$r = \sqrt{x_s^2 + y_s^2} \tag{17}$$

$$\alpha \overset{y_s \geq 0}{=} -\mathrm{arcsin}\frac{x_s}{r}. \tag{18}$$

Here r is the planar distance to the camera and α resembles the pan angle of the camera. Note that $\alpha = 0$ is aligned with the y-axis. To calculate the tilt angle, consider

$$\beta = \mathrm{arctan}\frac{c_z}{r} \tag{19}$$

for the right-angled triangle formed by c_z, r, and the distance between object and camera. To assemble a remap instruction for object coordinates to joint configuration, we resolve for x_s and y_s and obtain

$$x_s = r\sin(-\alpha) \tag{20}$$

$$y_s = \sqrt{c_z^2 - x_s^2}. \tag{21}$$

With (19), we get

$$x_s = \frac{c_z}{\tan(\beta)}\sin(-\alpha) \tag{22}$$

$$\begin{aligned} y_s^2 &= \frac{r^2}{\tan^2(\beta)} - \frac{c_z^2}{\tan^2(\beta)}\sin^2(-\alpha) \\ &= (1 - \sin^2(-\alpha))\frac{c_z^2}{\tan^2(\beta)}. \end{aligned} \tag{23}$$

Using $\sin^2 + \cos^2 = 1$, y_s is then

$$y_s = \frac{c_z}{\tan(\beta)}\cos(-\alpha). \tag{24}$$

For two joints with inverted angles $\mathrm{pan} = -\alpha$ and $\mathrm{tilt} = -\beta$, the final form is

$$x_s = -\frac{c_z}{\tan(\mathrm{tilt})}\sin(\mathrm{pan}) \tag{25}$$

$$y_s = -\frac{c_z}{\tan(\mathrm{tilt})}\cos(\mathrm{pan}). \tag{26}$$

Acknowledgment

The authors would like to thank J. P. Spencer from the University of Iowa for the extensive discussion and sharing of ideas, and S. Schneegans from the Institut für Neuroinformatik, University of Bochum, Germany.

References

[1] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Trans. Robot. Autom.*, vol. 17, no. 3, pp. 229–241, Mar. 2001.

[2] A. Maki, P. Nordlund, and J.-O. Eklundh, "Attentional scene segmentation: Integrating depth and motion from phase," *Comput. Vis. Image Understand.*, vol. 78, no. 3, pp. 351–373, 2000.

[3] A. Mishra, Y. Aloimonos, and C. Fermuller, "Active segmentation for robotics," in *Proc. IROS 2009*, St. Louis, MO, 2009, pp. 3133–3139.

[4] S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart, "Cognitive maps for mobile robots—An object based approach," *Robot. Autonom. Syst.*, vol. 55, pp. 359–371, 2007.

[5] A. Nüchter and J. Hertzberg, "Towards semantic maps for mobile robots," *Robot. Autonom. Syst.*, 2008.

[6] N. Vahrenkamp, A. Barski, T. Asfour, and R. Dillmann, "Planning and execution of grasping motions on a humanoid robot," in *Proc. IEEE-RAS Int. Conf. Human. Robot.—HUMANOIDS 2009*, Paris, France, 2009, pp. 639–645.

[7] R. Rusu, A. Holzbach, N. Blodow, and M. Beetz, "Fast geometric point labeling using conditional random fields," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, St. Louis, MO, 2009, pp. 7–12.

[8] C. Faubel and G. Schöner, "Learning to recognize objects on the fly: A neurally based dynamic field approach," *Neural Netw.*, vol. 21, pp. 562–576, 2008.

[9] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," *Comput. Vis. Image Understand.*, vol. 106, no. 1, pp. 59–70, 2007, 4.

[10] I. Iossifidis, C. Theis, C. Grote, C. Faubel, and G. Schöner, "Anthropomorphism as a pervasive design concept for a robotic assistant," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Las Vegas, NV, 2003, pp. 3465–3472.

[11] B. Bridgeman and M. Mayer, "Failure to integrate visual information from successive fixations," *Bulletin Psych. Soc.*, vol. 21, no. 4, pp. 285–286, 1983.

[12] D. Irwin, "Information integration across eye movements," *Cogn. Psychol.*, vol. 23, pp. 420–456, 1991.

[13] A. Treisman, "Feature binding, attention and object perception," *Philos. Trans. Roy. Soc. (London) B Biol. Sci.*, vol. 353, pp. 1295–1306, 1998.

[14] D. Irwin and R. Andrews, "Integration and accumulation of information across saccadic eye movements," *Attention Perform. XVI: Inform. Integr. Percept. Commun.*, pp. 125–155, 1996.

[15] R. Rensink, "The dynamic representation of scenes," *Vis. Cogn.*, vol. 7, pp. 17–42, 2000.

[16] J. O'Regan, R. Rensink, and J. Clark, "Blindness to scene changes caused by mudsplashes," *Nature*, vol. 34, p. 398, 1999.

[17] A. Hollingworth and J. Henderson, "Accurate visual memory for previously attended objects in natural scenes," *J. Exp. Psychol. Human Percept. Perform.*, vol. 28, no. 1, pp. 113–136, 2002.

[18] D. Kahneman, A. Treisman, and B. Gibbs, "The reviewing of object files: Object-specific integration of information," *Cogn. Psychol.*, vol. 24, no. 2, pp. 175–219, 1992.

[19] A. Hollingworth, "Memory for object position in natural scenes," *Vis. Cogn.*, vol. 12, no. 6, pp. 1003–1016, 2005.

[20] A. Hollingworth, "Object-position binding in visual memory for natural scenes and object arrays," *J. Exp. Psychol.: Human Percept. Perform.*, vol. 33, no. 1, pp. 31–47, 2007.

[21] W. Erlhagen and E. Bicho, "The dynamic neural field approach to cognitive robotics," *J. Neural Eng.*, vol. 3, p. R36, 2006.

[22] G. Schöner, "Dynamical systems approaches to cognition," in *Cambridge Handbook of Computational Cognitive Modeling*, R. Sun, Ed. Cambridge, U.K.: Cambridge Univ. Press, 2008, pp. 101–126.

[23] W. Erlhagen and G. Schöner, "Dynamic field theory of movement preparation," *Psychol. Rev.*, vol. 109, pp. 545–572, 2002.

[24] E. Thelen, G. Schöner, C. Scheier, and L. Smith, "The dynamics of embodiment: A field theory of infant perseverative reaching.," *Brain Behav. Sci.*, vol. 24, pp. 1–33, 2001.

[25] J. S. Johnson, J. P. Spencer, S. J. Luck, and G. Schöner, "A dynamic neural field model of visual working memory and change detection," *Psychol. Sci.*, vol. 20, pp. 568–577, 2009.

[26] V. R. Simmering, J. P. Spencer, and G. Schöner, "Reference-related inhibition produces enhanced position discrimination and fast repulsion near axes of symmetry," *Percept. Psychophys.*, vol. 68, pp. 1027–1046, 2006.

[27] A. R. Schutte, J. P. Spencer, and G. Schöner, "Testing the dynamic field theory: Working memory for locations becomes more spatially precise over development," *Child Develop.*, vol. 74, pp. 1393–1417, 2003.

[28] J. S. Johnson, J. P. Spencer, and G. Schöner, "Moving to higher ground: The dynamic field theory and the dynamics of visual cognition," *New Ideas Psychol.*, vol. 26, pp. 227–251, 2008.

[29] B. R. Fajen and W. H. Warren, "Behavioral dynamics of steering, obstacle avoidance, and route selection," *J. Exp. Psychol.: Human Percept. Perform.*, vol. 29, no. 2, pp. 343–262, 2003.

[30] G. Schöner, M. Dose, and C. Engels, "Dynamics of behavior: Theory and applications for autonomous robot architectures," *Robot. Autonom. Syst.*, vol. 16, pp. 213–245, 1995.

[31] E. Bicho, P. Mallet, and G. Schöner, "Target representation on an autonomous vehicle with low-level sensors," *Int. J. Robot. Res.*, vol. 19, pp. 424–447, 2000.

[32] G. Schöner and C. Engels, "Dynamic field architecture for autonomous systems," in *Proc. Percept. Action Conf.*, P. Gaussier and J.-D. Nicoud, Eds., Lausanne, Switzerland, Sep. 7–9, 1994, pp. 242–253, Los Alamitos, California: IEEE Computer Society Press, 1994.

[33] W. Erlhagen, A. Mukovskiy, E. Bicho, G. Panin, C. Kiss, A. Knoll, H. V. Schie, and H. Bekkering, "Action understanding and imitation learning in a robot-human task," *Lecture Notes Comput. Sci.*, vol. 3696, pp. 261–268, 2005.

[34] C. Faubel and G. Schöner, "Learning to recognize objects on the fly: A neurally based dynamic field approach," *Neural Netw. Special Issue Neurosci. Robot.*, vol. 21, no. 4, pp. 562–576, May 2008.

[35] I. Iossifidis and G. Schöner, "Autonomous reaching and obstacle avoidance with the anthropomorphic arm of a robotic assistant using the attractor dynamics approach," in *Proc. IEEE 2004 Int. Conf. Robot. Autom.*, New Orleans, LA, 2004, pp. 4295–4300.

[36] S. Amari, "Dynamics of pattern formation in lateral-inhibition type neural fields," *Biol. Cybern.*, vol. 27, pp. 77–87, 1977.

[37] J. Taylor, "Neural 'bubble' dynamics in two dimensions: Foundations," *Biol. Cybern.*, vol. 80, no. 6, pp. 393–409, 1999.

[38] K. Kopecz and G. Schöner, "Saccadic motor planning by integrating visual information and pre-information on neural, dynamic fields," *Biol. Cybern.*, vol. 73, pp. 49–60, 1995.

[39] C. Wilimzig, S. Schneider, and G. Schöner, "The time course of saccadic decision making: Dynamic field theory," *Neural Netw.*, vol. 19, pp. 1059–1074, 2006.

[40] P. S. Goldman-Rakic, "Cellular basis of working memory," *Neuron*, vol. 14, pp. 477–485, 1995.

[41] D. J. Amit, "The hebbian paradigm reintegrated: Local reverberations as internal representations," *Behav. Brain Sci.*, vol. 18, no. 4, pp. 617–626, 1994.

[42] V. R. Simmering, "Developing a Magic Number: The Dynamic Field Theory Reveals Why Visual Working Memory Capacity Estimates Differ Across Tasks and Development," Ph.D. dissertation, Dept. of Psychol., Univ. of Iowa, Iowa City, IO, 2008.

[43] J. Spencer and S. Perone, "A dynamic neural field model of multi-object tracking," *J. Vis.* vol. 8, no. 6, pp. 508–508, 2008 [Online]. Available: http://journalofvision.org/8/6/508/

[44] E. Dineva and G. Schöner, "Dynamic instabilities as mechanisms for emergence," *Develop. Sci.*, vol. 10, no. 1, pp. 69–74, 2007.

[45] V. Simmering, A. R. Schutte, and J. P. Spencer, "Generalizing the dynamic field theory of spatial cognition across real and developmental time scales," *Brain Res.*, vol. 1202, pp. 68–86, 2008.

[46] Y. Sandamirskaya and G. Schöner, "Dynamic field theory and embodied communication," in *Modeling Communication for Robots and Virtual Humans*, ser. Springer Lecture Notes in Computer Science/Lecture Notes in Artificial Intelligence, I. Wachsmuth and G. Knoblich, Eds. Berlin, Germany: Springer-Verlag, 2008, vol. 4930, pp. 260–278.

[47] J. S. Johnson, J. P. Spencer, and G. Schöner, "A dynamic neural field theory of multi-item visual working memory and change detection," in *Proc. 28th Annu. Conf. Cogn. Sci. Soc. (CogSci 2006)*, Vancouver, BC, Canada, 2006, pp. 399–404.

[48] C. Wilimzig and G. Schöner, "The emergence of stimulus-response associations from neural activation fields: Dynamic field theory," in *Proc. 27th Annu. Conf. Cogn. Sci. Soc.*, Stresa, Italy, 2005, pp. 2359–2364.

[49] Y. Sandamirskaya and G. Schöner, "Dynamic field theory of sequential action: A model and its implementation on an embodied agent," in *Proc. Int. Conf. Develop. Learn. ICDL'2008*, B. Scassellati and G. Deak, Eds., 2008, p. 8.

[50] C. Faubel and G. Schöner, "A neuro-dynamic architecture for one shot learning of objects that uses both bottom-up recognition and top-down prediction," in *Proc. IEEE/IRSJ Int. Conf. Intell. Robot. Syst.*, St. Louis, MO, 2009, pp. 3162–3169.

[51] A. Pouget and T. Sejnowski, "Spatial transformations in the parietal cortex using basis functions," *J. Cogn. Neurosci.*, vol. 9, no. 2, pp. 222–237, 1997.

[52] R. Andersen, R. Bracewell, S. Barash, J. Gnadt, and L. Fogassi, "Eye position effects on visual, memory, and saccade-related activity in areas LIP and 7a of macaque," *J. Neurosci.*, vol. 10, no. 4, p. 1176, 1990.

[53] K. Tanaka, "Neuronal mechanisms of object recognition," *Science*, vol. 262, pp. 685–688, 1993.

[54] J. Schall and D. Hanes, "Neural basis of saccadic target selection in frontal eye field during visual search," *Nature*, vol. 366, pp. 467–469, 1993.

[55] D. L. Sparks, "Conceptual issues related to the role of the superior colliculus in the control of gaze," *Current Opinion Neurobiol.*, vol. 9, no. 6, pp. 698–707, 1999.

[56] D. Sparks, "The brainstem control of saccadic eye movements," *Nature Rev. Neurosci.*, vol. 3, no. 12, pp. 952–964, 2002.

[57] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[58] V. Braitenberg, *Vehicles. Experiments in Synthetic Psychology*. Cambridge, MA: MIT Press, 1984.

[59] P. Dudek and P. Hicks, "A general-purpose processor-per-pixel analog SIMD vision chip," *IEEE Trans. Circuit. Syst. I*, vol. 52, no. 1, pp. 13–20, Jan. 2005.

[60] J. Fix, J. Vitay, and N. Rougier, "A distributed computational model of spatial memory anticipation during a visual search task," *Anticipatory Behav. Adapt. Learn. Syst.* pp. 170–188, 2007 [Online]. Available: http://dx.doi.org/10.1007/978-3-540-74262-3_10

[61] C. Choi and H. I. Christensen, "Cognitive vision for efficient scene processing and object categorization in highly cluttered environments," in *Proc. IEEE/IRSJ Int. Conf. Intell. Robot. Syst. IROS*, St. Louis, MO, 2009, pp. 4267–4274.

[62] A. Dankers, N. Barnes, W. Bischof, and A. Zelinsky, On Real-Time Synthetic Primate Vision 2007.

[63] R. Rensink, J. O'Regan, and J. Clark, "On the failure to detect changes in scenes across brief interruptions," *Vis. Cogn.*, vol. 7, no. 1, pp. 127–145, 2000.

[64] J. Lipinski, Y. Sandamirskaya, and G. Schöner, "Flexible spatial language behaviors: Developing a neural dynamic theoretical framework," in *Proc. 9th Int. Conf. Cogn. Model.*, Manchester, U.K., 2009, pp. 257–264.

[65] J. Lipinski, Y. Sandamirskaya, and G. Schöner, "Behaviorally flexible spatial communication: Robotic demonstrations of a neurodynamic framework," KI 2009: Advances in Artificial Intelligence B. Mertsching, Ed., 2009.

[66] A. Steinhage and G. Schöner, "Dynamical systems for the behavioral organization of autonomous robot navigation," in *Proc. Sensor Fusion Decentralized Contr. Robot. Syst.: Proc. SPIE*, M. G. T. Schenker P. S., Ed., 1998, vol. 3523, pp. 169–180, SPIE-publishing, no. ISBN 0-8194-2984-8.

Stephan K. U. Zibner received the B.Sc and M.Sc. degrees from the Faculty of Electrical Engineering and Information Sciences, University of Bochum, Bochum, Germany. He is currently working towards the Ph.D. degree in the Autonomous Robotics Group at the Institut für Neuroinformatik, Ruhr-Universität Bochum, Bochum, Germany.

He studied Applied Computer Science with a focus on media and communication.

Christian Faubel studied Industrial Technoloy, for which he received a prediploma from Paul Sabatier University, Toulouse III, France. He continued his studies, receiving a diploma in Mechanical Engineering at Dresden University of Technology, Dresden, Germany. He received the Ph.D. degree from the Faculty of Electrical Engineering and Information Sciences, University of Bochum, Bochum, Germany.

He is currently a Postdoctoral Researcher in the Autonomous Robotics Group at the Institut für Neuroinformatik, Ruhr-Universität Bochum, Bochum, Germany. His research focuses on cognitive models of fast robotic learning of object representations.

Ioannis Iossifidis received the Dipl. degree in physics from the University of Dortmund, Dortmund, Germany and the Ph.D. degree from the Faculty of Physics and Astronomy of the University of Bochum, Bochum, Germany.

He is currently the Director of the Cognitive System Technology Lab and holds the Chair for Theoretical Computer Science at the Hochschule Ruhr West Mühlheim. His research focuses on movement generation for redundant robot arms, on modeling of human movements, and learning sequences in cognitive robotic systems on the basis of dynamical systems ideas.

Gregor Schöner received the Dipl. degree in physics from the University of Saarbrücken, Saarbrücken, Germany, in 1982, and the Ph.D. degree in theoretical physics from the University of Stuttgart, Stuttgart, Germany, in 1985.

He is currently the Director of the Institut für Neuroinformatik at the Ruhr-Universität Bochum, Bochum, Germany, and also holds the chair for Theoretical Biology. For the last 25 years, he has brought to bear his background in theoretical physics on interdisciplinary research in cognitive science, neuroscience, kinesiology, robotics, and computer vision. While working closely with experimental groups, he has developed theoretical concepts and models to understand movement, perception, and elementary forms of cognition on the basis of dynamical systems ideas. He has held academic positions in the United States, France, and Germany, has published over 170 scientific articles, and frequently lectures all over the world.